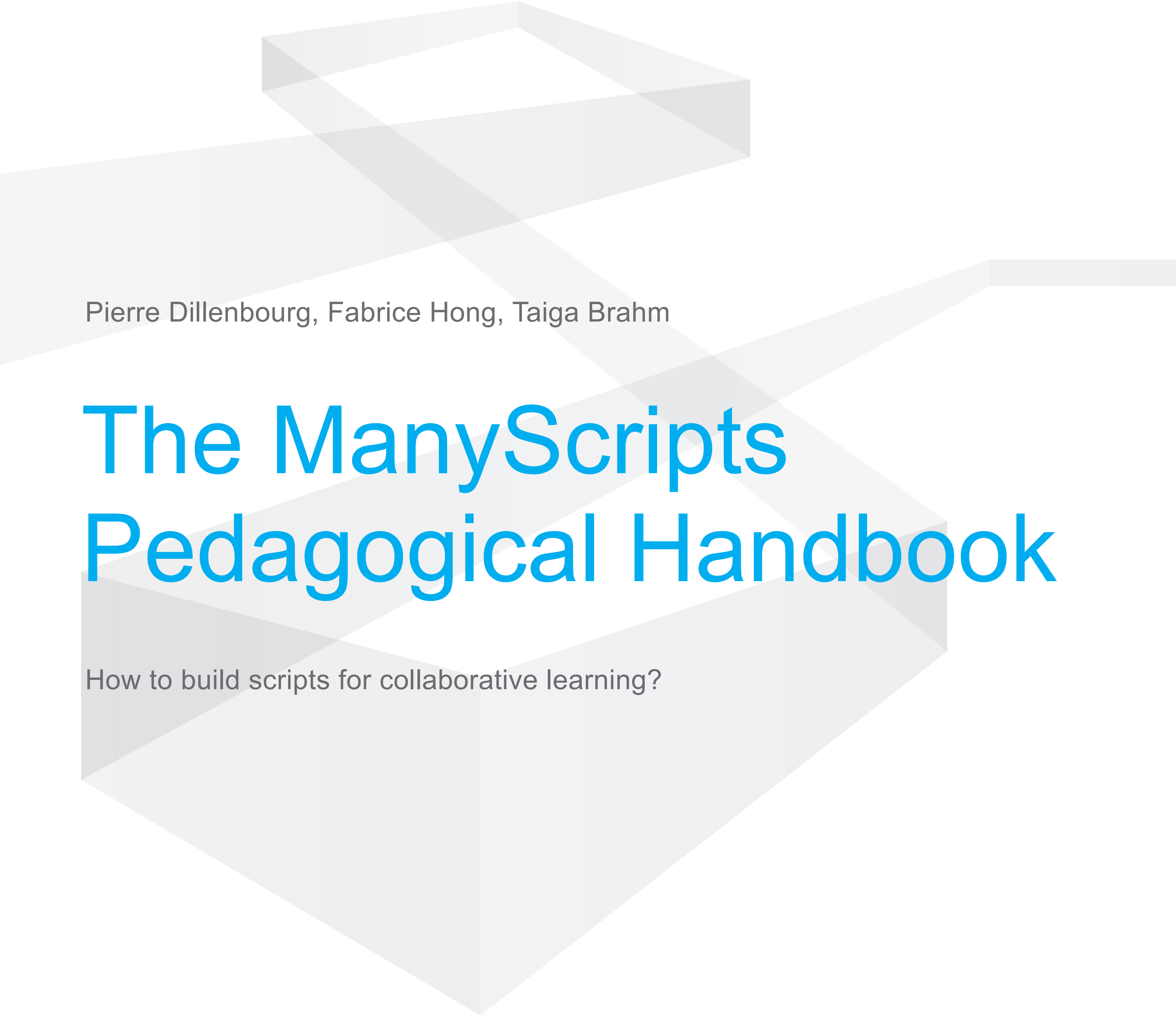
scil report 20
June 2009



Pierre Dillenbourg, Fabrice Hong, Taiga Brahm

The ManyScripts Pedagogical Handbook

How to build scripts for collaborative learning?

Institute of Business Education
and Educational Management
University of St.Gallen

scil, swiss centre for
innovations in learning

Initiated and Supported by the Gebert Rüf Foundation

scil report 20
June 2009

Pierre Dillenbourg, Fabrice Hong, Taiga Brahm

The ManyScripts Pedagogical Handbook

How to build scripts for collaborative learning?

ISBN: 978-3-906528-58-8

Institute of Business Education
and Educational Management
University of St.Gallen

scil, swiss centre for innovations in learning

Initiated and Supported by the Gebert Rüf Foundation

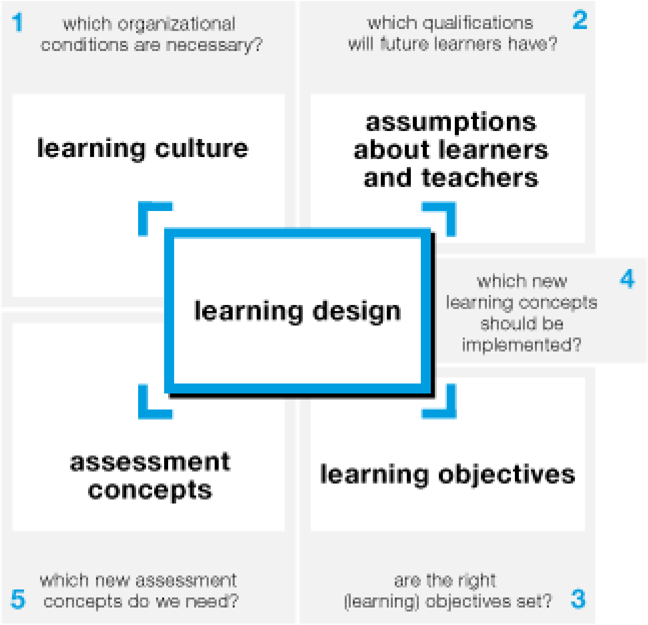
Content

Foreword 4
1. Introduction........ 6
2. The ManyScripts Environment 10
3. The ArgueGraph script 11
4. The ConceptGrid script 25
5. The IceGrain script 34
6. The IceCube script 45
7. Managing scripts 47
Literature 52

Foreword

The Swiss Centre for Innovations in Learning (scil) provided research and development fellowships in order to establish international cooperation and development innovations in the field of learning. From 2005 to 2008, one of the scil fellows was Pierre Dillenbourg, professor of pedagogy and learning technologies in the School of Computer and Communication Sciences at the Swiss Federal Institute of Technology in Lausanne (EPFL).

This scil fellowship was part of scil's research and development activities in the area of "learning design" (see following figure).


1 which organizational conditions are necessary?
learning culture
2 which qualifications will future learners have?
assumptions about learners and teachers
learning design
4 which new learning concepts should be implemented?
assessment concepts
learning objectives
5 which new assessment concepts do we need?
3 are the right (learning) objectives set?

One major question for innovations in learning design is which new learning concepts should be implemented in higher education and in companies in the future (no 4 in the figure). Computer-supported collaborative learning (CSCL), and in particular scripting as a possibility to support learning processes, are possible concepts to be implemented. Collaborative learning can be defined as "a *situation* in which *two or more* people *learn* or attempt to learn something *together*" (Dillenbourg, 1999, p 1).

Since collaborative learning is not always effective, a closer look needs to be taken at how interaction during CSCL can be made richer, more intense and more meaningful. One approach is to script the collaboration process through specific phases, roles, and activities. This also includes the forming of groups in a specific way. During the fellowship, four specific scripts were developed:

- ArgueGraph,
- ConceptGrid,
- IceCube,
- IceGrain.

All four scripts are integrated in a comprehensive support environment, called ManyScripts (see chapter 2).

The following scil report is a pedagogical handbook on the usage of these scripts. In chapter 1, the concepts of collaborative learning and scripting are introduced, followed by an overview on the ManyScripts environment (chapter 2). Chapters 3 to 6 provides in-depth introduction to the four scripts developed and a guidance on how to implement them in the classroom. The last chapter concludes with notes on how to manage scripts from an instructor's point of view (chapter 7).

In sum, the report emphasizes that structuring and scaffolding of collaborative learning is needed. The principles introduced in the four scripts can be transferred to other computer-supported learning environments such as the usage of web 2.0-tools like wikis and blogs (see also Seufert & Brahm, 2007; Cress & Kimmerle, 2008).

1. Introduction

What is collaborative learning?

Collaborative learning is the process of working together on a given topic with the aim to reach certain learning objectives. It concerns groups of two or more people, working face-to-face or on-line, for one hour or several years. Co-learners may work rather independently in some phases and more interactively in others, the whole process being still described as collaborative.

What is a script?

A pedagogical method.

A collaborative script is a pedagogical scenario that students have to follow when they learn together. Instead of free collaboration, a script structures the collaboration process e.g. by prescribing different activities or by instructing how to form the group. Some of these activities are computer-based, some are not.

What is ManyScripts?

A tool for authoring and running scripts.

ManyScripts is a web-based environment where teachers may prepare the script they want to use with their students. Later on, the student will login to ManyScripts to do the activities that compose the script. ManyScripts is somewhat similar to a learning management system such as Moodle, but focused on a few pedagogical methods called scripts.

Why scripting?

To increase effectiveness.

Empirical studies show that collaborative learning is often more effective than learning alone, often but not always! Some groups do not work well together and hence do not learn much. The effectiveness of teamwork depends on the richness and intensity of interactions in which the group members engage. For instance, if the teacher asks students to argue about an issue, some groups will engage into deep arguments, raising the key issues that the teacher expects them to address, while other groups will remain at a superficial level, repeating common places. While the former team will benefit from this activity, the latter will not learn much.

How can we make sure that both teams will argue intensively? Actually, there is no method that "guarantees" effective collaborative learning. Nonetheless, some well-*designed* activities increase the probability of positive outcomes. For instance, in an argumentation activity, forming groups of students who have conflicting opinions is a "design" feature that increases students' engagement. If students do not have opposite opinions, another "design" choice for inducing conflicting opinions is to provide them with different documents to read, each document containing evidence for opposite viewpoints. One may also ask them to play the role of different characters known for their opposing viewpoints. These are examples of tricks that teachers may use to "design" the way their students will work together. A script integrates these tricks within a pedagogical scenario, i.e. a sequence of activities.

What is the pedagogical idea behind a script?

There are two principles.

The first principle is that free collaboration is not always productive and hence that defining some *structure* for activities will scaffold collaborative processes. The degree of structuring required for learning is a debate as old as education: the lack of structure leads to unproductive activities during which students do not learn anything; however, too much structure can also impede students' learning experience. The right amount of structure varies according to the students' knowledge level and is expected to decrease as learning unfolds.

The second principle is that scripts do not make collaboration easier but somehow more difficult. For instance, as ArgueGraph forms pairs of students with opposite opinions, it is for them more difficult to agree on responses than if pairs would be formed with students of similar opinions. Hence, the script requires students to argue more intensively, to explain to each other, etc. Tuning this *additional effort* – not too low, not too high- is the art of defining collaborative scripts.

Are scripts only for collaborative learning?

No.

Scripts do not only include collaborative learning activities but also individual activities and collective activities. Individual activities are, for instance, reading a paper before teamwork or writing a summary after teamwork. In collective activities, all students from the class are gathered with the teacher, for instance, for an introductory lecture or for a debriefing session. During such a session, the teacher will ask students to compare their solutions, to comment upon each others' contributions and to articulate their reflections to the contents of the course. Debriefing sessions are the cornerstone of most scripts: team activities provide students with a meaningful experience but individual or group reflection is necessary to crystallize these insights, *to turn experience into learning.*

Why using computers ?

It does not really matter!

Our scripts rely on a software environment. Using computers for running scripts has both advantages and drawbacks. The drawbacks are the management of computer access for all students and the difficulty to modify scripts once teachers have initiated them (some features can be changed, but not all). The advantages mostly concern the logistics of the scripts: the ICE script enables 50 students to share and criticize documents; the ArgueGraph automatically forms pairs of students having opposite opinions. We used computerized scripts in contexts where the advantages outweigh the disadvantages. When it was the other way around, we designed scripts to be run without computers. With or without computers is not a question anymore, the question is to implement the pedagogical design relevant to the learning objectives.

Are scripts designed for distance teaching?

No.

Even if they rely on computers, our scripts are not designed for distance education but for enhancing classroom activities. Scripts include activities that can be done online, for instance at home, but the key activities are designed for situations where all students meet in a room with their teacher. Variations of the script could be created for distance courses, but it this not the case yet.

When teachers think about using the internet in their courses, they consider providing online documents and selected links, a discussion forum, some simulation applets, etc. Scripts broaden the spectrum of activities that bring the benefits of Internet into the classroom.

Can I take holidays during the script?

No, the teacher has a key role in "orchestrating" the scenario.

Some activities may occur independently from teachers, but the timing of the sequence, the composition of groups and the nature of assignments often need to be adapted on the fly, due to the numerous unexpected events that can occur during a course.

More importantly, the role of the teacher is central to the collective activities, especially the debriefing activities, and for each activity that requires feedback. Teachers have instead to expect a rather high workload during the script period.

How long does a script run?

It depends on the script contents.

Typically, an ArgueGraph script with 10 questions will take about four hours, but, of course, this time will vary proportionally to the number of questions. An ICE script may last anywhere between two hours and six months. Finding the right timing is indeed a difficult aspect of scripts.

Can I use a script for teaching any topic?

No.

Each script is relevant for some learning objectives. The issue is not whether scripts would be more relevant for teaching biology than for teaching mathematics, but more or less relevant for different learning objectives within a domain. For instance, within the same course, an ArgueGraph script could be used for chapter 1 and ICE for chapter 5. This handbook describes the relevance of each script for a set of learning objectives.

Can I use a script with young students?

Not as they stand now.

The scripts presented in this handbook have been designed for university courses and tested in this context. Some activities such as reading scientific papers do not transfer easily to younger students without being adapted. The pedagogical principles of the 3 presented scripts would apply to lower school levels but they certainly require some adaptation.

Can I use a script for corporate training?

Yes.

We used these scripts in university courses but also in various seminars with colleagues. For corporate training, we recommend to use scripts in blended learning courses so that individual phases can be conducted on-line and team or class phases can be run face-to-face. For instance, the best way to exploit the short time available in a residential seminar would be that students run the ArgueGraph phases 1 and 2 online, before the seminar, do phases 3 and 4 during the seminar and complete phase 5 after the seminar. For the ConceptGrid, all phases but the last one, could be run before the seminar. The debriefing activities are very productive ways to use seminar times.

Summary:

- Scripts aim at triggering rich interactions in the teams.
- A script is a sequence of activities or phases.
- Some activities are individual, some in small groups and some with the whole class
- Some activities use computers, some don't, but the whole script is managed by ManyScripts.epfl.ch

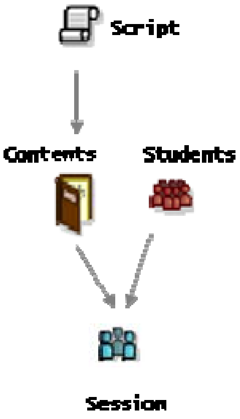
2. The ManyScripts Environment

4 available scripts:

- ArgueGraph
- ConceptGrid
- IceCube
- IceGrain

ManyScripts is a web-based platform that strives for simplicity: it is not a powerful authoring tools where users may build new scripts, but a small library that, despite its pretentious name, includes only 4 scripts. We focused on making these few scripts effective with the hope that more scripts will be developed in the future.

Preparing a script in 3 steps:



1. Defining the script **contents**. First, the author selects one of the proposed scripts. A script describes the phases, the activities, and the group formation, defined in a way independent from the contents of the script. To define the contents, the teacher will, for instance, enter the questions students should answer or the papers they should read. She also has to choose some parameters such as group size.
2. Defining the **class** of students that will use the script. The teacher may enter all student names or let them register by themselves.
3. Defining the **session** consists in associating a script content to a class of student and determining the timing of each phase. The session will store all elements that are specific to these students, for instance all answers or contributions made by students in various activities.

Example

A biology teacher could create two instances of ArgueGraph; one on cell division and one on genetics. He is teaching to two classes called class 3 (20 math students) and class 4 (50 biology students). He would prepare 3 sessions: class 3 will use the script on cells in January while class 4 will use both scripts instances, the one about cell division in February and the one about genetics in June.

What do you need to run it?

ManyScripts runs on our servers at the Swiss Federal Institute of Technology in Lausanne. To obtain an account, please contact pierre.dillenbourg@epfl.ch. Teachers and students only need a browser. ManyScripts has been tested on Firefox and Internet Explorer. Some functionalities may not work on other browsers.

Is it bug free ?

The ArgueGraph and ConceptGrid scripts have been used several times and should be mostly bug free. The IceGrain and IceCube scripts were only tested once in a real context. ManyScripts remains a prototype which cannot be compared to a commercial product.

Sharing your work

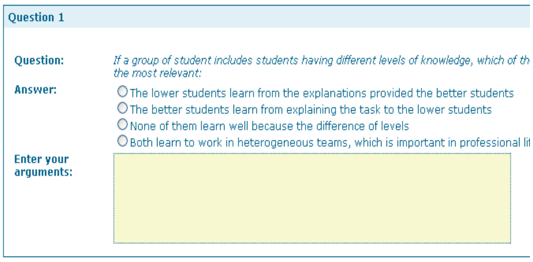
Teachers may share scripts with colleagues who can hence reuse the contents created. This is explained in chapter 7.

3. The ArgueGraph script

Overview of the script

The aim of ArgueGraph is to trigger argumentation between peers. This is achieved by collecting opinions and forming pairs of students with opposite opinions. The script includes 5 phases that we briefly describe now and will then develop phase by phase.

Phase 1. Individual questionnaire



Question 1

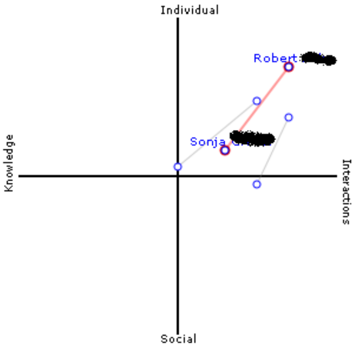
Question: *If a group of student includes students having different levels of knowledge, which of th the most relevant:*

Answer:
○ The lower students learn from the explanations provided the better students
○ The better students learn from explaining the task to the lower students
○ None of them learn well because the difference of levels
○ Both learn to work in heterogeneous teams, which is important in professional lif

Enter your arguments:

Each student responds to an online multiple choice questionnaire. These questions do not have right or wrong answers, but rather reflect different viewpoints. These viewpoints are to be addressed by the teacher in Phase 4. For each answer the student is expected to write a few lines justifying his or her choice.

Phase 2. Group formation



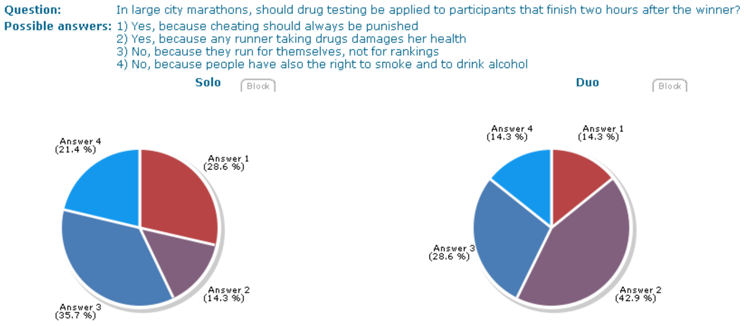
Based on the responses in Phase 1, the system produces a map of opinions. The teacher discusses this map with students and forms pairs in such a way that distance between students is maximized, i.e. pairing of students who provided conflicting responses in Phase 1.

Phase 3: Pair argumentation

Pairs answer the same questionnaire as in Phase 1. The environment displays the answers and justifications provided by each peer in Phase 1. Pairs must select a single answer and write a few lines to justify their choice.

Phase 4: Class discussion

This debriefing session aims at reformulating the elements mentioned by the students using the correct terminology, to structure them and to integrate them into a theoretical framework. The teacher synthesizes the elements provided by the students, asks them to provide further clarification, rephrase their justifications, compare them and so on.



Question: In large city marathons, should drug testing be applied to participants that finish two hours after the winner?

Possible answers:
1) Yes, because cheating should always be punished
2) Yes, because any runner taking drugs damages her health
3) No, because they run for themselves, not for rankings
4) No, because people have also the right to smoke and to drink alcohol

Phase 5: Individual summary

Each student chooses one of the questions discussed and writes a summary of all elements collected by the system in Phases 1 and 3, structured according to the framework developed in Phase 4.

Choosing ArgueGraph

Learning objectives

The main objective is that the argumentation processes forces students to elicit their own knowledge. The goal is not that students learn to argue with each other. Argumentation is an effective method to elicit what one knows and the ArgueGraph is a way to increase argumentation. An analysis of previous experiments showed that conflict led pairs to produce justifications that were not proposed by any peer in the individual phase

Conceptual overview

We have used ArgueGraph in courses where we had to browse an overview of a conceptual domain that has been developed around different school of thoughts. Through ArgueGraph, students elicited most of the concepts and principles that would have been presented in a lecture. However, these elements were produced without choosing the right names for concepts, without structuring them into theoretical approaches. The principles that students used to justify their choices are based on personal experience rather than on robust evidence. The teacher's role is to turn this magma into a structured conceptual map during the debriefing phase (phase 4).

The ArgueGraph can be used in inquiry learning (e.g. Gijlers & DeJong, 2005). Each question could address a scientific phenomenon and the answers to the question would be different hypotheses that can be formulated about this phenomenon. The student would then "guess" the best hypothesis in the solo phase but "prove" it in the duo phase by using a computer-based simulation or a real experimental set-up.

Many topics

ArgueGraph is not limited to subjective fields where personal opinions dominate, such as artistic or political domains. Students are not expected to argue about whether they like something or not but to defend or attack a statement based on rational statements. ArgueGraph is hence relevant to scientific and engineering domains. For instance, students may be asked to argue about 3 different methods to estimate noise in network communication, about three possible medical diagnoses for a set of symptoms or three weather forecasts for the same barometric map.

Class size: up to 50

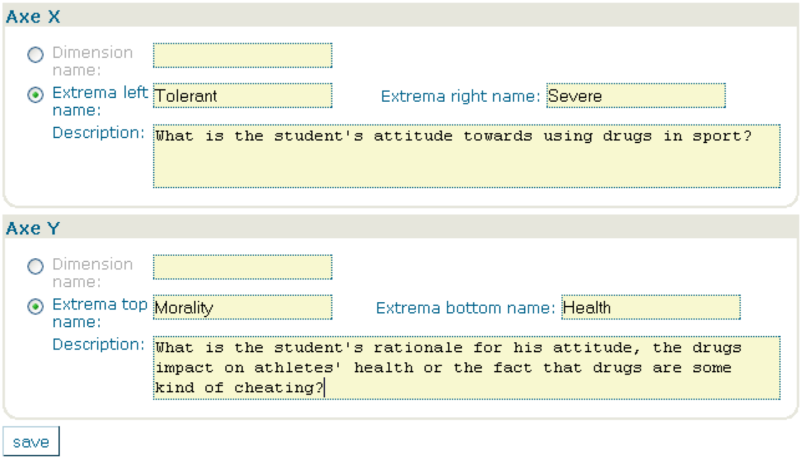
We recommend to use ArgueGraph with classes ranging from 6 to 50 students. Technically speaking, the system may accommodate many more students. The main limitation to the class size is due to the interactivity of the debriefing phase: from time to time, every student will be asked to defend his or her choices by the teacher, which limits the class size to about 40-50 students.

Creating the script content

The first step is to define the two axes of the opinions map used in phase 2. These axes must be defined before entering the questions. The author may either label the axes or give names to each end. The description field is only for documenting the script for reuse.

(Step 1) Defining axes

For the system, these two axes are just two labels, but for the effectiveness of the script, these axes must be chosen carefully. The choice of these two axes will determine how the teacher will exploit the students' answers during the debriefing session (phase 4), i.e. when these answers have to be related to the theoretical framework. Therefore, the axes have to reflect the theoretical space addressed in the course, i.e. to discriminate the main theoretical approaches, to reveal the key differences between these approaches, to point out which areas have been more or less explored, to show the evolution of the field.



Axe X

○ Dimension name:

● Extrema left name: Tolerant

Extrema right name: Severe

Description: What is the student's attitude towards using drugs in sport?

Axe Y

○ Dimension name:

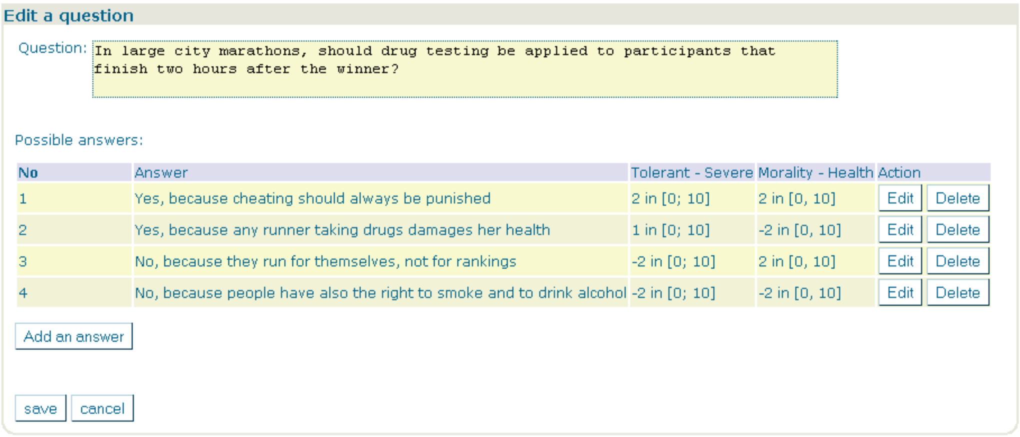
● Extrema top name: Morality

Extrema bottom name: Health

Description: What is the student's rationale for his attitude, the drugs impact on athletes' health or the fact that drugs are some kind of cheating?

(Step 2) Writing questions

The second and most sensitive step when is to choose the set of questions used in phases 1 and 3. The author will define a set of questions and for each question a set of answers. Typically, 10 questions with 3 to 4 answers each will feed a 4 hours ArgueGraph sessions. For each answer, the author will specify a pair of values such as [2 0] that will be used to locate students on the 2 axes that have been defined. Values must be an integer between 0 and 10.



Edit a question

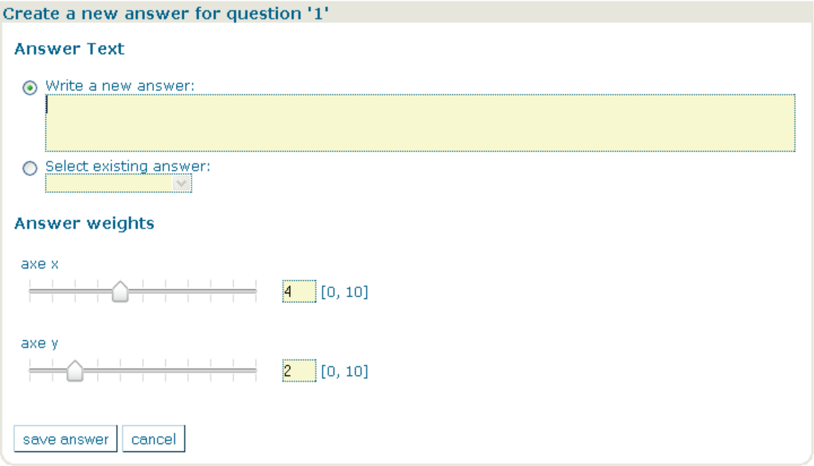
Question: In large city marathons, should drug testing be applied to participants that finish two hours after the winner?

Possible answers:

No	Answer	Tolerant - Severe	Morality - Health	Action
1	Yes, because cheating should always be punished	2 in [0; 10]	2 in [0, 10]	Edit Delete
2	Yes, because any runner taking drugs damages her health	1 in [0; 10]	-2 in [0, 10]	Edit Delete
3	No, because they run for themselves, not for rankings	-2 in [0; 10]	2 in [0, 10]	Edit Delete
4	No, because people have also the right to smoke and to drink alcohol	-2 in [0; 10]	-2 in [0, 10]	Edit Delete

Multiple correct answers

Questions have to *trigger* argumentation. First of all, question should not have one correct answer but rather *multiple correct answers* depending on the theoretical viewpoint chosen, which are the theoretical viewpoints that the teacher wants to illustrate. "Is Geneva in the west of Lausanne?" will trigger no argumentation; a good question must have equally plausible answers. "Should we allow free use of drugs at the Olympics?" will not generate much argumentation because some consensus exists in our society. A question such as "Should we allow athletes to inject before Olympics their own blood collect in winter time?" will lead to diverging answers.



Create a new answer for question '1'

Answer Text

● Write a new answer:

○ Select existing answer:

Answer weights

axe x

4 [0, 10]

axe y

2 [0, 10]

No compromise

For the same reasons, the different answers should be *contrasted*. Answers including words that add nuances, such as "in general" and "in some cases" will enable consensus among peers without any argumentation. Ordinal answer scales such as ["I strongly agree", "I agree", "I am neutral", "I disagree", "I strongly disagree"] or [never rarely sometimes often always] will also produce soft consensus and hence kill argumentation.

Distribution on the 2 axes

Most questions should contain answers that correspond to each different quadrants defined by the two axes, but this is not always possible.

Equally plausible answers

The set of answers aims at *spreading* the set of students over the map used in phase 2 and hence has to be formulated so that convergence is avoided. We recommend to avoid answers that have a "politically correct" flavour; otherwise all students end up in the same quarter of the graph.

Referring to course contents

Last but not least, the arguments entered by students when selecting an answer will feed the debriefing phase. Hence, the choice of question needs to anticipate the fact that the arguments produced by students will refer to the *content elements* (concepts, principles, …) that the teacher wants to integrate in the theoretical framework to be interactively constructed in phase 4 .

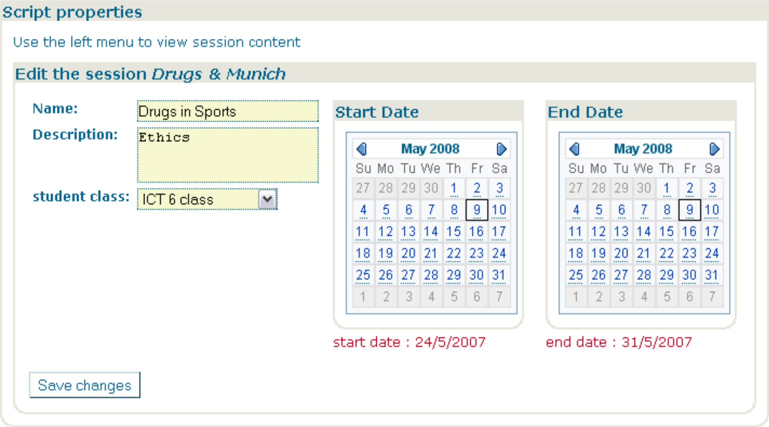
Order of answers

As in any multiple-choice question, the order of answers must vary across questions: the answer that corresponds for instance the top-right quadrant of the map should not come always come first, etc.

Defining the ArgueGraph session:

Let us assume at this point that the teacher has defined a script content and a class of students. To create a session, the script content is associated to a class and the relevant dates are defined.

(step 3) script timing



Script properties

Use the left menu to view session content

Edit the session *Drugs & Munich*

Name: Drugs in Sports

Description: Ethics

student class: ICT 6 class

Start Date

May 2008

Su	Mo	Tu	We	Th	Fr	Sa
27	28	29	30	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31
1	2	3	4	5	6	7

start date : 24/5/2007

End Date

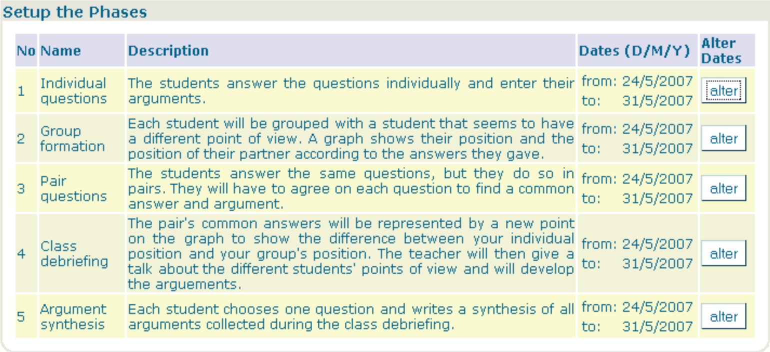
May 2008

Su	Mo	Tu	We	Th	Fr	Sa
27	28	29	30	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31
1	2	3	4	5	6	7

end date : 31/5/2007

Similarly, the dates of each phase have to be defined. This is not very relevant for the ArgueGraph script, which often runs in a few hours, but very important for scripts that stretch over several weeks.

(step 4) phases timing



Setup the Phases

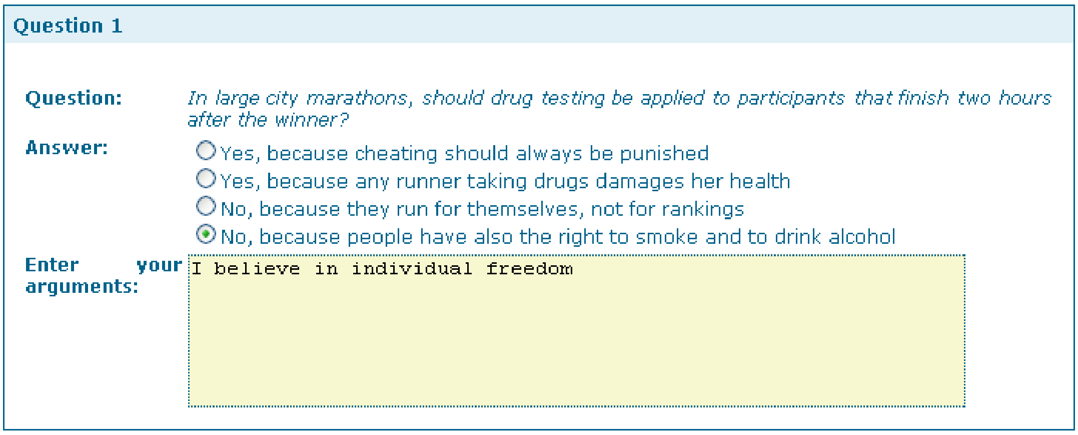
No	Name	Description	Dates (D/M/Y)	Alter Dates
1	Individual questions	The students answer the questions individually and enter their arguments.	from: 24/5/2007 to: 31/5/2007	alter
2	Group formation	Each student will be grouped with a student that seems to have a different point of view. A graph shows their position and the position of their partner according to the answers they gave.	from: 24/5/2007 to: 31/5/2007	alter
3	Pair questions	The students answer the same questions, but they do so in pairs. They will have to agree on each question to find a common answer and argument.	from: 24/5/2007 to: 31/5/2007	alter
4	Class debriefing	The pair's common answers will be represented by a new point on the graph to show the difference between your individual position and your group's position. The teacher will then give a talk about the different students' points of view and will develop the arguements.	from: 24/5/2007 to: 31/5/2007	alter
5	Argument synthesis	Each student chooses one question and writes a synthesis of all arguments collected during the class debriefing.	from: 24/5/2007 to: 31/5/2007	alter

Running Phase 1:

Individual Questionnaire

About 10 minutes

In phase 1, each student has to individually answer each question. Typically, this will take 20 minutes for 10 questions with 4 possible answers. If the number of computers available is lower than the number of students, this may be done in successive waves but in this case the number of questions should be reduced.



Question 1

Question: *In large city marathons, should drug testing be applied to participants that finish two hours after the winner?*

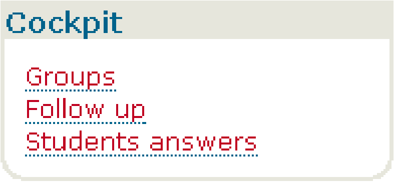
Answer:

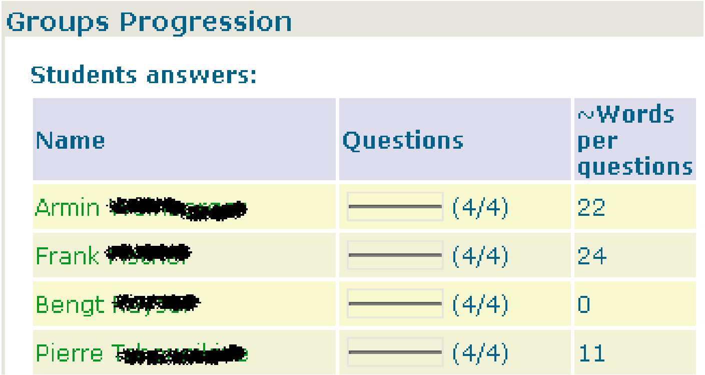
○ Yes, because cheating should always be punished
○ Yes, because any runner taking drugs damages her health
○ No, because they run for themselves, not for rankings
● No, because people have also the right to smoke and to drink alcohol

Enter your arguments: I believe in individual freedom

Teacher's follow-up

The teacher has to manage the diversity of individual speed so that students complete phase 1 more or less together. All students must answer all questions. Some students will have finished before others, they may already see a partial map. The role the teachers is to ask them to wait for the others and at the same time to push the others to complete the questionnaire. Therefore, the ManyScripts 'cockpit' menu provides tools to follow the students' activity. The option 'Follow-Up' shows how many questions have been completed by how many students and hence help to manage time.





Groups Progression

Students answers:

Name	Questions	~Words per questions
Armin [redacted]	(4/4)	22
Frank [redacted]	(4/4)	24
Bengt [redacted]	(4/4)	0
Pierre [redacted]	(4/4)	11

Write short justifications!

Students have to justify why they choose an answer. Usually, most students need to be encouraged to write more than 2 words. The teacher has to repeat a few times "please justify your choices". However, a few students sometimes enter very long justifications which will delay the rest of the class. We recommend to tell students explicitly that a justification should be 2 lines long. The students' interface has been designed to induce this length. The cockpit column "Words per question" enables the teacher to detect students who enter too short or too long justifications.

Endless hesitations

Students may complain that none of the answers corresponds to their opinion and, for that reason, some of them even refuse to answer some questions. In this case, we explain that a script is just a didactic game, i.e. that there is no problem if the answer does not exactly match their opinion, that any answer is needed to proceed. We then encourage them to choose the answer they dislike the least and to specify in the justification text how their own answer would be different.

Saving work

There is a SAVE button at the bottom of the questionnaire. Students' responses need to be saved manually so that, if their computer crashes or if they lose the connection, they may simply login again and proceed where they were. Teachers should remind them from time to time to save their work.

Moving to phase 2

If the teacher does not want to wait for the last answers of the last students, two situations are possible:

The teacher wants to keep students with incomplete answers involved in the next phases (e.g. a student refused to answer one question), the teacher may then simply proceed. The horizontal and vertical points of missing answers will simply not be counted.

The teacher wants to proceed without this student (e.g. a student dropped out), then he should simply remove the student from the class (section "manage students class", see chapter 6).

If a student does not answer any question at all, (s)he will simply not be considered by the group formation algorithm.

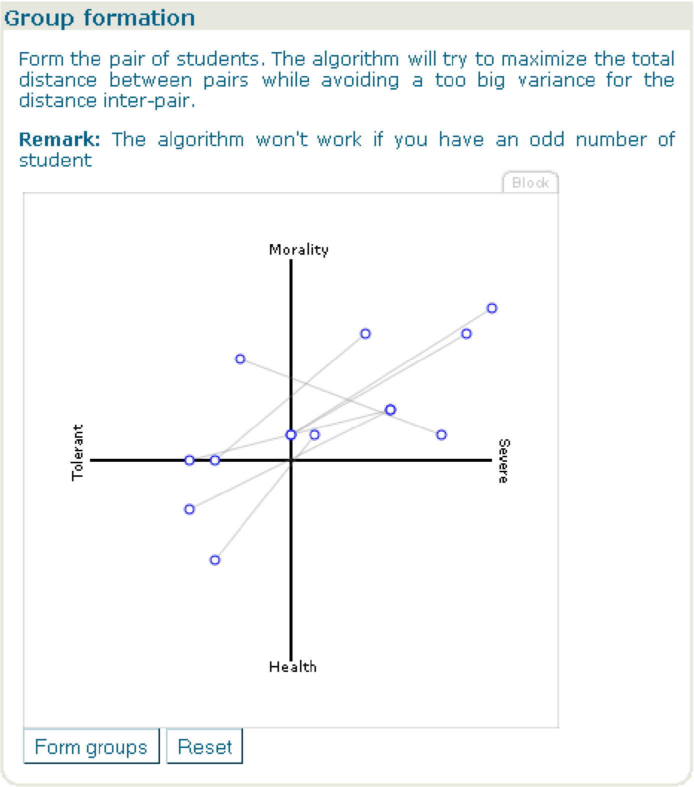
Running Phase 2:

Group formation

In the second phase, ArgueGraph produces a map reflecting all answers. The location of each student on the graph is computed as the sum of the [x y] values associated with each of his or her answers. The names of the axes are those defined by the teacher.

The teacher should stress that this map does not provide a scientifically accurate picture of each opinion. It is a didactic artefact that has no validity beyond the fact that it raises interest among students and supports the next steps of the script.

ArgueGraph produces a map based on students' answers



Group formation

Form the pair of students. The algorithm will try to maximize the total distance between pairs while avoiding a too big variance for the distance inter-pair.

Remark: The algorithm won't work if you have an odd number of student

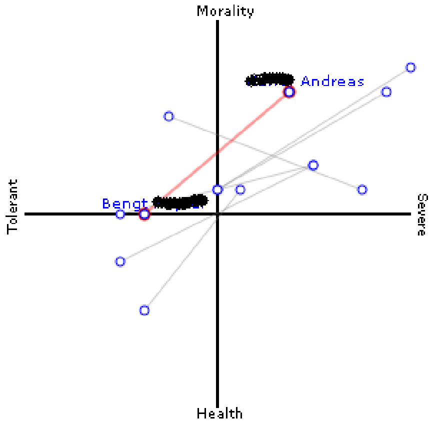
First impression of the map

This map always triggers many reactions in the class. Students compare their position to those of others, express surprise ("I never thought I would be close to you") and try spontaneously to explain positions and distances. There is a risk that someone located far away from the rest of the group might feel hurt by a map revealing his or her isolation. It has not happened in any of the sessions we did but teachers should be aware of this risk. The informal discussion around this map typically lasts for about 5 minutes.

ArgueGraph forms pairs of students who are far from each other on the map

The teacher has to click on the "Form groups" button. ManyScripts then forms pairs in a way that maximizes the average distance between peers, i.e. the average divergence of opinions. If there is an odd number of students, the group formation algorithm will form pairs and has to place the last student manually. The results of group formation are displayed as lines on the social map. Moving the mouse over the links displays the names of the students who have to work together in phase 3.

Map of pairs



Manual group formation

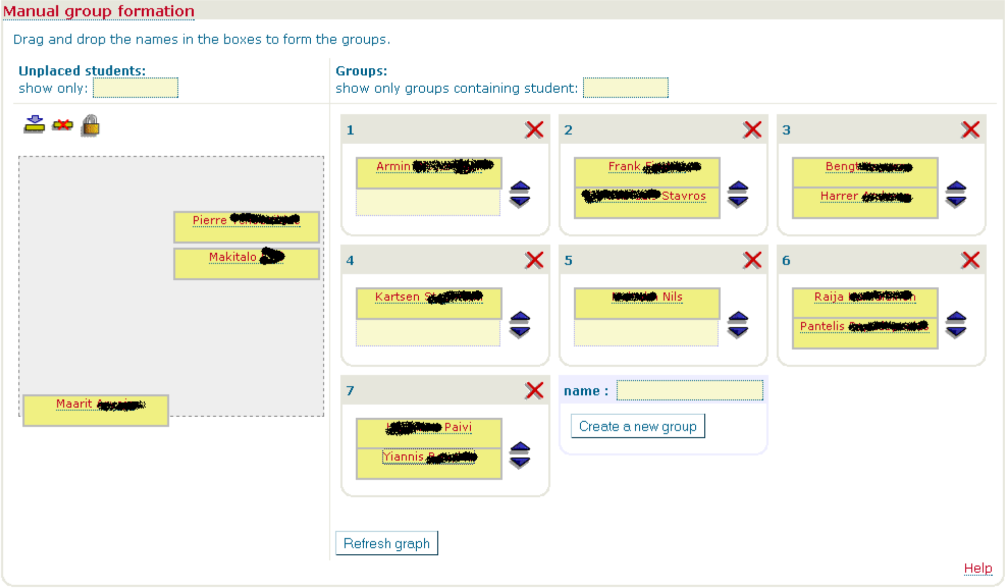
In some cases, the teacher may want to form the pairs by himself. The students are displayed on the left grey pane. The teacher creates groups of two in the right hand side pane and then drags and drops students from the left list to the right groups.

Combining automatic and manual group formation

In some cases, the teacher may want to modify the pairs formed by the system. Manual group formation can be combined with automatic group formation: the algorithm first forms the groups that the teacher may next manually modify. This is especially useful if there is an odd number of students: the teacher may then add one seat in a group (by clicking on the ▼ button) and drag the last student to that group. This feature is supported to cope with special cases, however, the whole ArgueGraph interface is designed for pairs, not for larger groups. Other functionalities of the group formation window are explained in chapter 3 (section "running ArgueGraph phase 1").

Beware!

Changing groups after the start of phase 2 will delete their group answers !



Manual group formation

Drag and drop the names in the boxes to form the groups.

Unplaced students:
show only:

Groups:
show only groups containing student:

Pierre [redacted]

Makitalo [redacted]

Maarit [redacted]

1: Armin [redacted]

2: Frank [redacted] / [redacted] Stavros

3: Bengt [redacted] / Harrer [redacted]

4: Kartsen [redacted]

5: [redacted] Nils

6: Raija [redacted] / Pantelis [redacted]

7: [redacted] Paivi / Yiannis [redacted]

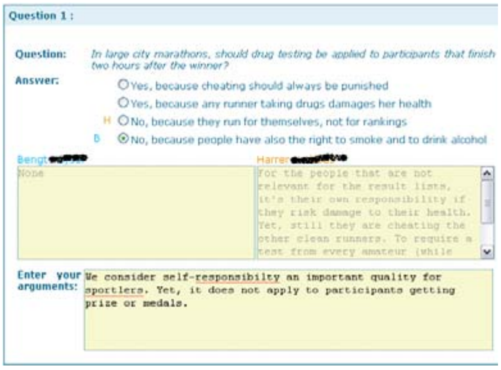
name :

Running Phase 3:

Pair argumentation

About 40 minutes

The newly formed pairs sit together in front of a computer. One of the two members login. The system displays the name of each pair member. For each question, the individual answers are indicated (the initial of each student in front of the answer line) and the justifications of each member are displayed.



Question 1 :

Question: *In large city marathons, should drug testing be applied to participants that finish two hours after the winner?*

Answer:
○ Yes, because cheating should always be punished
○ Yes, because any runner taking drugs damages her health
H ○ No, because they run for themselves, not for rankings
B ● No, because people have also the right to smoke and to drink alcohol

Bengt [redacted]
None

Harrer [redacted]
for the people that are not relevant for the result lists, it's their own responsibility if they risk damage to their health. Yet, still they are cheating the other clean runners. To require a test from every amateur (while

Enter your arguments: We consider self-responsibilty an important quality for sportlers. Yet, it does not apply to participants getting prize or medals.

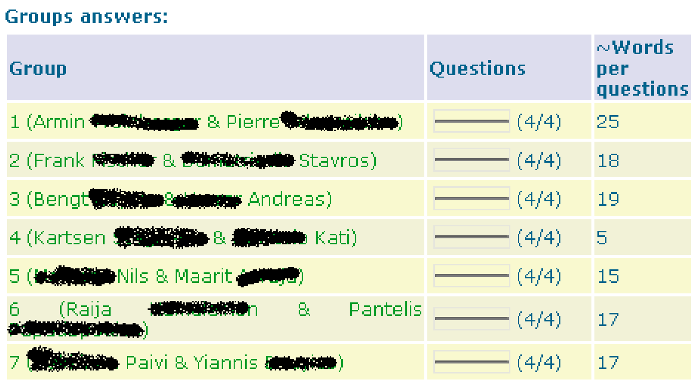
Usually, the room gets very noisy: the level of noise is a good approximation of the intensity of argumentation. This phase typically lasts twice as long as phase 1, i.e. about 40 minutes for 10 questions. While the team set of answers is often a mid-way between the two sets of individual answers, as mentioned above, we also see cases where their argumentation leads them to positions beyond individual ones.

The teacher's role is similar to phase 1: managing teams so that they complete the questionnaire more or less at the same time. It occurs that some teams do not manage to agree on one of the proposed answers and argue forever. When it really takes too long, the teacher has to convince them to choose one answer, even if they disagree, just to be able to move on. They may still express their disagreement in the justifications.

Managing time

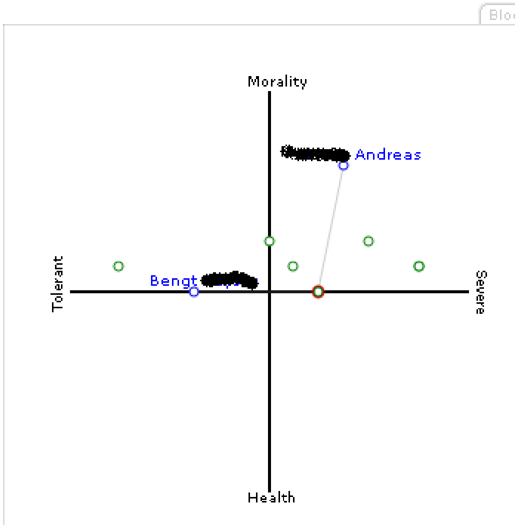
The frustration that some students express for not finding an answer that matches what they would like to answer is actually not a bad thing: in general, these students participate even more to the debriefing session because they then have the opportunity to say publicly what they could not express in the questionnaire.

As for phase 1, the teacher may monitor the evolution of pair response and at the end may display how the pair answer differs from the answers of each of the members. The red dot below locates the point answers of Andreas and Bengt while the blue dots indicate their individual answers.



Groups answers:

Group	Questions	~Words per questions
1 (Armin [redacted] & Pierre [redacted])	(4/4)	25
2 (Frank [redacted] & [redacted] Stavros)	(4/4)	18
3 (Bengt [redacted] & [redacted] Andreas)	(4/4)	19
4 (Kartsen [redacted] & [redacted] Kati)	(4/4)	5
5 ([redacted] Nils & Maarit [redacted])	(4/4)	15
6 (Raija [redacted] & Pantelis [redacted])	(4/4)	17
7 ([redacted] Paivi & Yiannis [redacted])	(4/4)	17



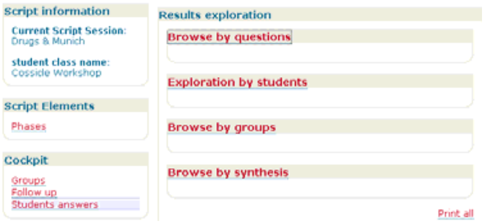
Running Phase 4:

Class debriefing

Duration: 1 or 2 hours

Phase 4 is the most challenging for the teacher. His or her task is to discuss the answers given in the previous phases in order to organize them into a consistent framework. The difficulty is that the session cannot be prepared in a detailed way but has be constructed on the fly. We often included a 15 minutes-coffee break between phases 3 and 4 giving us the opportunity to have a glimpse on students' answers.

This session benefits from the energy accumulated by students in phase 3, which leads them to engage with some passion into the discussion of their answers. Namely, students who could not express precisely their opinions are very motivated to complete their answers during the open debriefing session. Once we did this debriefing session one week after phase 3 and this energy was lost, students had forgotten what and why they answered. Hence, keep the break between phases 3 and 4 short.



Script information

Current Script Session: Drugs & Munich

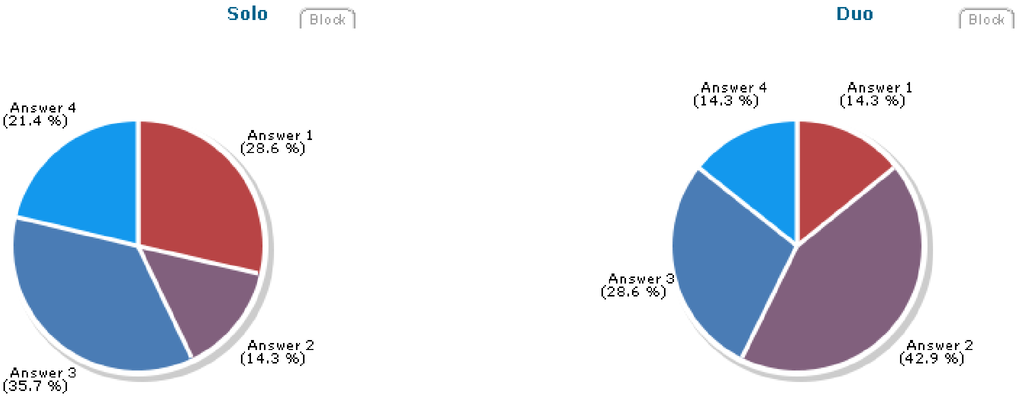
student class name: Cosside Workshop

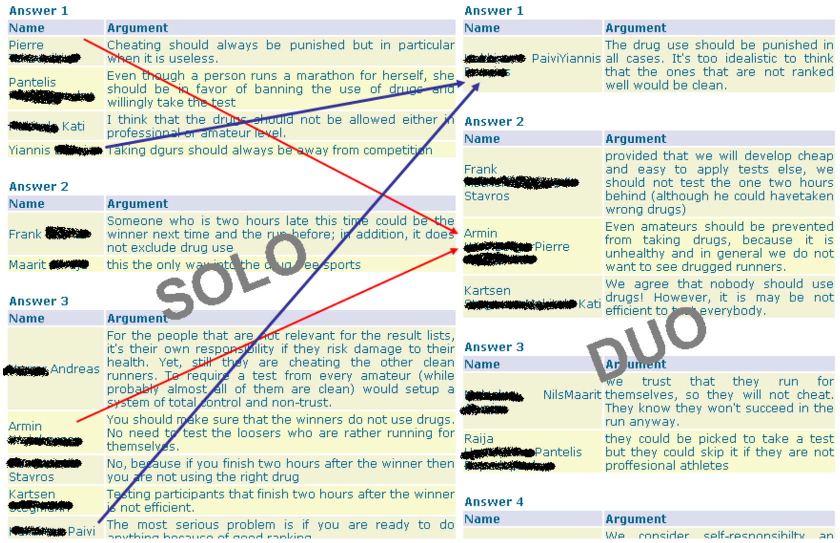
The teacher explores in real time the answers provided by the students (in the cockpit, under "students' answers"). The answers can be explored in different ways but we suggest to explore them question by question in sequential order (option "browse by question" in the menu "students' answers" of the Teacher's Cockpit- see below)..

The debriefing is not a feedback session in a traditional sense. The point is not to indicate who gave right answers since there is in principle no right answer. The goal is to re-organize the answers by relating them to a theoretical framework. We usually display the following charts on the projector and discuss them briefly. For each question, the tool provides a pie chart of individual answers (on the left) and pair answers (on the right). We usually display this graph for a few seconds and then analyse the list of their justifications (see next page).

Question: In large city marathons, should drug testing be applied to participants that finish two hours after the winner?

Possible answers:
1) Yes, because cheating should always be punished
2) Yes, because any runner taking drugs damages her health
3) No, because they run for themselves, not for rankings
4) No, because people have also the right to smoke and to drink alcohol





Answer 1

Name	Argument
Pierre [redacted]	Cheating should always be punished but in particular when it is useless.
Pantelis [redacted]	Even though a person runs a marathon for herself, she should be in favor of banning the use of drugs and willingly take the test
[redacted] Kati	I think that the drugs should not be allowed either in professional or amateur level.
Yiannis [redacted]	Taking dgurs should always be away from competition

Answer 2

Name	Argument
Frank [redacted]	Someone who is two hours late this time could be the winner next time and the run before; in addition, it does not exclude drug use
Maarit [redacted]	this the only way into the drug free sports

Answer 3

Name	Argument
[redacted] Andreas	For the people that are not relevant for the result lists, it's their own responsibility if they risk damage to their health. Yet, still they are cheating the other clean runners. To require a test from every amateur (while probably almost all of them are clean) would setup a system of total control and non-trust.
Armin [redacted]	You should make sure that the winners do not use drugs. No need to test the loosers who are rather running for themselves.
[redacted] Stavros	No, because if you finish two hours after the winner then you are not using the right drug
Kartsen [redacted]	Testing participants that finish two hours after the winner is not efficient.
[redacted] Paivi	The most serious problem is if you are ready to do anything because of good ranking

Answer 1

Name	Argument
[redacted] PaiviYiannis [redacted]	The drug use should be punished in all cases. It's too idealistic to think that the ones that are not ranked well would be clean.

Answer 2

Name	Argument
Frank [redacted] Stavros	provided that we will develop cheap and easy to apply tests else, we should not test the one two hours behind (although he could havetaken wrong drugs)
Armin [redacted]Pierre [redacted]	Even amateurs should be prevented from taking drugs, because it is unhealthy and in general we do not want to see drugged runners.
Kartsen [redacted]Kati	We agree that nobody should use drugs! However, it is may be not efficient to test everybody.

Answer 3

Name	Argument
[redacted] NilsMaarit [redacted]	we trust that they run for themselves, so they will not cheat. They know they won't succeed in the run anyway.
Raija [redacted]Pantelis [redacted]	they could be picked to take a test but they could skip it if they are not proffesional athletes

Answer 4

Name	Argument
	We consider self-responsibilty an

Examples of questions during the debriefing phase

- Who changed his mind between the solo and duo phases? For instance, in the results above, answer 2 went from the least to the most selected. Why?
- What made you change your mind? For instance, in the figure above, we would ask Paivi how she has been convinced by Yannis to choose answer 1 while she answered 3 individually. We would ask Pierre and Armin why they choose together an answer that none of them had chosen individually.
- Why did answer 1 correspond to a different location on the map than answer 2? What is fundamentally different between answers 1 and 2?
- Which answer would you add to this question that would be closer to your opinions? How is it different from the proposed answers?
- Which of these 4 opinions is the most widespread in the class, in the society, in the press,…? Why ?
- Do you know any empirical evidence that supports one specific answer? Which kind of empirical evidence could help choosing among these different answers?
- What is the key element in the justification that you have written? Can we summarize the different justifications for the same answer into a more general point? What is the relationship between the points that justify answer B in question 1 and answer C in question 5?
- Do these answers reflect different theoretical approaches? Which answers corresponds to theoretical viewpoints we identified in the previous questions? Do different theoretical viewpoints lead to similar answers?

Running Phase 5 (optional):

Individual summary

The script ends with an optional phase in which students have to write an individual summary of the arguments brought up by all students during the preceding phases. The summary should be entered online. The tool displays to the student the list of elements used so far in the argumentation (see below). Typically, this is homework to be delivered one or two weeks later. We recommend to grade this summary or to consider it as exam material.

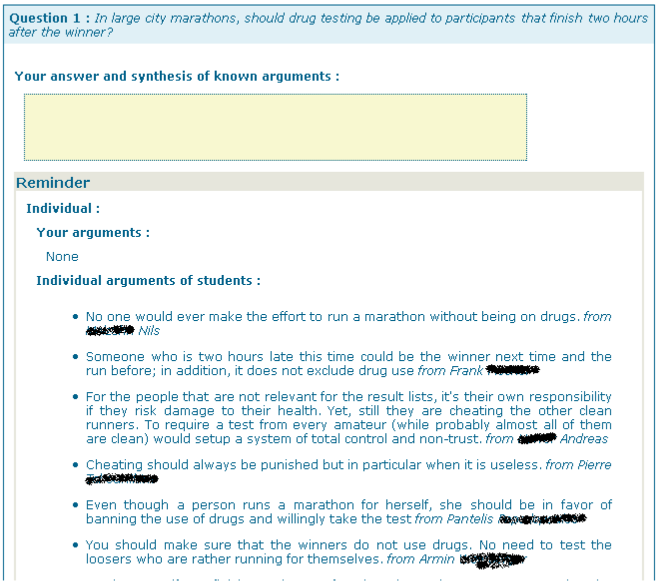
Duration: 1 or 2 weeks

Example of instructions

"You have to write a summary of the arguments that came up in the whole class. Please select one question among those you had to answer and sum up what has been said about it. Your summary must include two parts. In the first part, the justifications used by (most of) the students should be rephrased with the concepts

given to students

introduced by the teacher and classified into different theoretical frameworks. Different types of empirical evidence were mentioned in the lecture and should be referred to. In the second part, give your personal opinion and explain why some the elements weigh more on your opinion"



Question 1 : *In large city marathons, should drug testing be applied to participants that finish two hours after the winner?*

Your answer and synthesis of known arguments :

Reminder

Individual :

Your arguments :

None

Individual arguments of students :

- No one would ever make the effort to run a marathon without being on drugs. *from [redacted] Nils*
- Someone who is two hours late this time could be the winner next time and the run before; in addition, it does not exclude drug use *from Frank [redacted]*
- For the people that are not relevant for the result lists, it's their own responsibility if they risk damage to their health. Yet, still they are cheating the other clean runners. To require a test from every amateur (while probably almost all of them are clean) would setup a system of total control and non-trust. *from [redacted] Andreas*
- Cheating should always be punished but in particular when it is useless. *from Pierre [redacted]*
- Even though a person runs a marathon for herself, she should be in favor of banning the use of drugs and willingly take the test *from Pantelis [redacted]*
- You should make sure that the winners do not use drugs. No need to test the loosers who are rather running for themselves. *from Armin [redacted]*

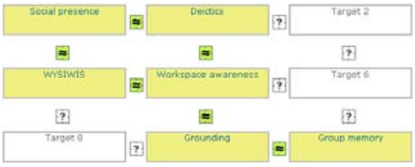
ArgueGraph without computers ?

ArgueGraph can also be run without computers. We tested the following version for a class of 20 students. We printed 30 copies of the questionnaire

1. Students answer the individual questionnaire on paper.
2. The teacher displays the horizontal value for each answer and students add their individual score to obtain their X location on the graph. The same operation is repeated for the Y value. After that the students shout their name and their [X,Y] value to the teacher who plots them manually on the map. Then, the teacher forms pairs manually by looking at the map.
3. The pairs formed in phase 2 answer the questionnaire on paper, still having their individual answer sheets with them
4. In the debriefing, the teacher does not see the students' answers and justifications, he must ask them to communicate them

4. The ConceptGrid script

Overview of the script



ConceptGrid belongs to a set of scripts referred to as Jigsaws: each team member only get pieces of the Jigsaw, so that he or she cannot complete it without sharing his or her knowledge with others. In the ConceptGrid, the students members acquire pieces of knowledge by reading different papers. Then, they have to define concepts and assemble them in a grid in such a way that they can explain the relationship between each grid neighbour. The concept grid can only be constructed if each team member explains the concepts about which he/she has read individually.

Phase 1

The students or the teacher form groups of, for instance, 3 students

Phase 2

Each group has to play 3 roles. It distributes the roles among its members. Each role is associated with papers to read.

Phase 3

Each student reads the papers associated with his or her role

Phase 4

The group distributes the concepts to be defined among its members.

Phase 5

Each student enters a 5-10-line definition of the concepts he or she has chosen to define.

Phase 6

The group constructs a concept grid, i.e. concepts are ordered on a map in such a way that two neighboring concepts can be explained in just a few sentences (see below).

Phase 7

This debriefing session aims at reformulating the definitions and relations provided by the students, to structure them and to integrate them into a theoretical framework.

Choosing a ConceptGrid:

Learning objectives

The goal of the ConceptGrid is that students learn about a theoretical domain by acquiring concepts and relating these concepts with each other. The target is declarative knowledge. ConceptGrid is suited for courses where students are not familiar with the field yet, e.g. students of computer science who discover human-computer interaction. ConceptGrid is also one way to force students to read papers, although some of them may look for definition on Internet instead of reading papers.

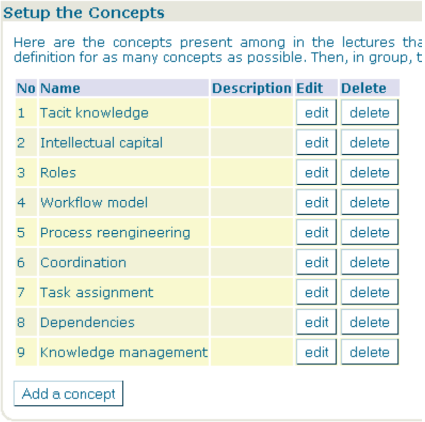
Class size

We have used ConceptGrid in master courses with 9 to 30 students. Larger classes raise difficulties for the debriefing phase. In a course, we ran 4 successive ConceptGrid scripts along the semester, one every 3 weeks, on different chapters. Students had to complete the grid the day before the lecture so that we were able to integrate the students' grids into the next day presentation. The grids were not graded but the students' incentive to produce them on time was that we they were printed on that day and these printed documents were available to students during the exam.

Creating the script content

A ConceptGrid requires little content. The key step is to decide the concepts that students will have to define and to assemble in the grid. The author simply enters the names of the concepts. It is possible to add a short description: this is only necessary if different concepts have the same name in closely related fields and the teacher wants to make sure students pick the right definition.

(Step 1) Choosing the concepts



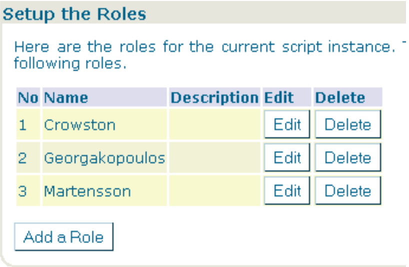
Setup the Concepts

Here are the concepts present among in the lectures tha
definition for as many concepts as possible. Then, in group, t

No	Name	Description	Edit	Delete
1	Tacit knowledge		edit	delete
2	Intellectual capital		edit	delete
3	Roles		edit	delete
4	Workflow model		edit	delete
5	Process reengineering		edit	delete
6	Coordination		edit	delete
7	Task assignment		edit	delete
8	Dependencies		edit	delete
9	Knowledge management		edit	delete

- The concepts must be probably unknown by most students, otherwise defining them does not require any reading and building the grid is too easy.
- The concepts must be rather difficult so that students cannot guess their meaning but have to engage into mutual explanations.
- "False friends" concepts are especially interesting: these are concepts that sound very similar to each other but are actually not, such as "democratic" and "election": the discrimination will be stressed during the debriefing phase.
- "Far neighbours" concepts are also interesting: these are concepts taken from different domains but nonetheless referring to a similar abstract notion: the common abstract notion will be stressed during the debriefing phase.
- The number of concepts is typically 3-4 per member, i.e. 8-16 per team. More concepts would make the grid construction too complex. If the number of concepts is a multiple of the number of students per team, the script encourages division of labour in phase 4 and 5. In the opposite case, students have to negotiate the way they distribute concepts among themselves.

(Step 2) Choosing roles



Setup the Roles

Here are the roles for the current script instance. ˉ
following roles.

No	Name	Description	Edit	Delete
1	Crowston		Edit	Delete
2	Georgakopoulos		Edit	Delete
3	Martensson		Edit	Delete

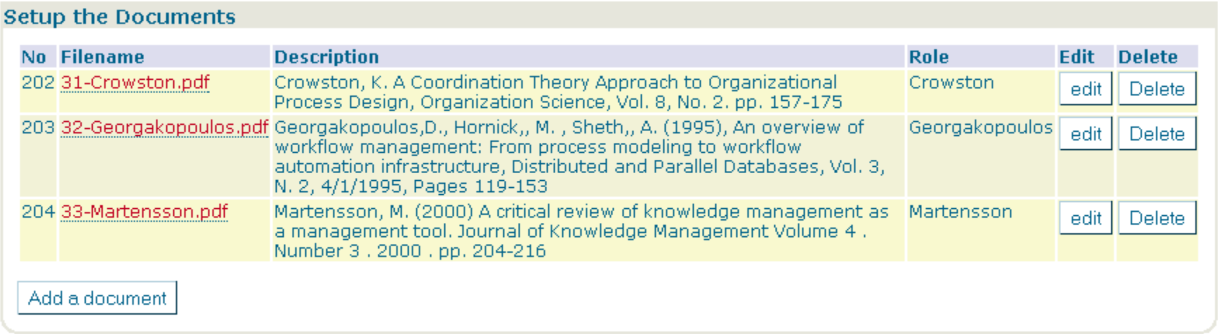
The teacher has to choose the readings necessary to define the concepts. This occurs in two steps. In the ideal case, the author defines roles such as "Piaget" and, to become Piaget or Pasteur, the student has to read 3 papers from Piaget or about Piaget. We applied this in postgraduate courses where intensive readings are more common. In undergraduate teaching, we often restrict to one paper.

By defining the number of roles, the teacher automatically defines the group size. We recommend groups of 3 to 4 students. Groups of 2 are somewhat too small for this activity while groups over 4 will have many difficulties in building the grid, which is often done by sitting together in front of a computer.

(Step 3) Choosing the papers to read

The teacher should upload papers as PDF or text file. As participants are authenticated before logging into the session, making documents available fits with the intellectual property rules that apply to schools in many countries. If the course combines several ConceptGrid sessions, one alternative is to print all papers in one volume such as "Readings in…".

Choosing the right papers is a critical step. They are supposed to be broader than most available research papers but also more scientific than most vulgarisation texts. Moreover, other pragmatic constraints have unfortunately to be taken into account such as the size of the paper (some interesting overviews are 60 pages long) and the language. It took me usually one hour to find each paper.



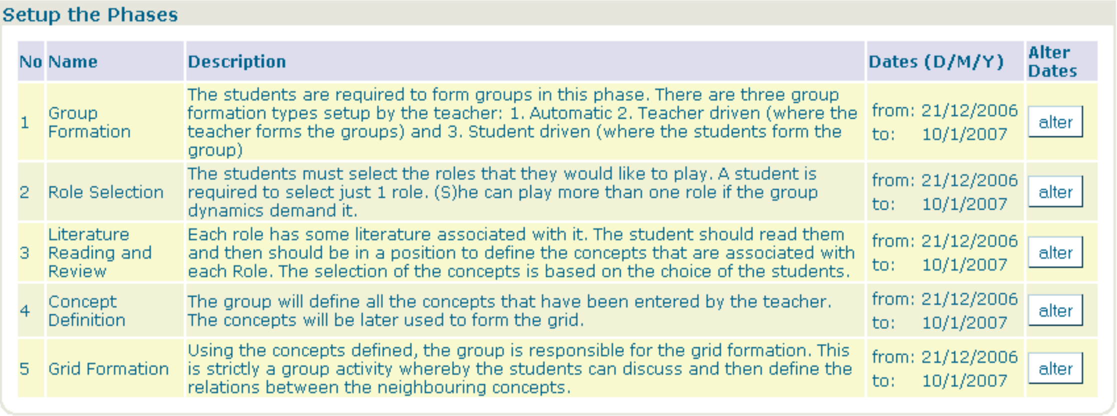
Setup the Documents

No	Filename	Description	Role	Edit	Delete
202	31-Crowston.pdf	Crowston, K. A Coordination Theory Approach to Organizational Process Design, Organization Science, Vol. 8, No. 2. pp. 157-175	Crowston	edit	Delete
203	32-Georgakopoulos.pdf	Georgakopoulos,D., Hornick,, M. , Sheth,, A. (1995), An overview of workflow management: From process modeling to workflow automation infrastructure, Distributed and Parallel Databases, Vol. 3, N. 2, 4/1/1995, Pages 119-153	Georgakopoulos	edit	Delete
204	33-Martensson.pdf	Martensson, M. (2000) A critical review of knowledge management as a management tool. Journal of Knowledge Management Volume 4 . Number 3 . 2000 . pp. 204-216	Martensson	edit	Delete

Defining the session:

(Step 4) Timing

The timing is important as a ConceptGrid is often spread over more than one week. Two other options are important: the grid dimensions and the way to form groups.

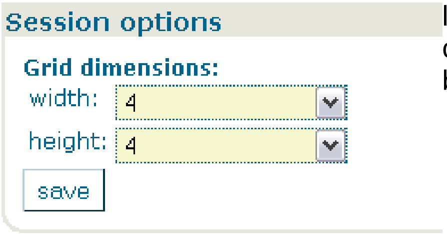


Setup the Phases

No	Name	Description	Dates (D/M/Y)	Alter Dates
1	Group Formation	The students are required to form groups in this phase. There are three group formation types setup by the teacher: 1. Automatic 2. Teacher driven (where the teacher forms the groups) and 3. Student driven (where the students form the group)	from: 21/12/2006 to: 10/1/2007	alter
2	Role Selection	The students must select the roles that they would like to play. A student is required to select just 1 role. (S)he can play more than one role if the group dynamics demand it.	from: 21/12/2006 to: 10/1/2007	alter
3	Literature Reading and Review	Each role has some literature associated with it. The student should read them and then should be in a position to define the concepts that are associated with each Role. The selection of the concepts is based on the choice of the students.	from: 21/12/2006 to: 10/1/2007	alter
4	Concept Definition	The group will define all the concepts that have been entered by the teacher. The concepts will be later used to form the grid.	from: 21/12/2006 to: 10/1/2007	alter
5	Grid Formation	Using the concepts defined, the group is responsible for the grid formation. This is strictly a group activity whereby the students can discuss and then define the relations between the neighbouring concepts.	from: 21/12/2006 to: 10/1/2007	alter

(step 5) Defining the grid

Our original idea was that all grid cells should be filled with concepts, but once, due to an error, we asked students to put 9 concepts on a 4 X 4 grid and they appreciated it very much - probably simply because it is easier. In this case, some concepts may have a single neighbour. If the semantic field is very consistent, i.e. each concept could be related with another one, we recommend a number of cells equal to the number of concepts (below left). In an uneven semantic field,



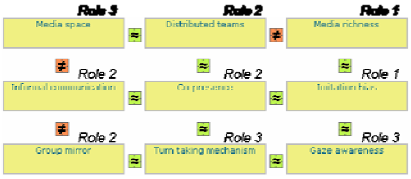
Session options

Grid dimensions:

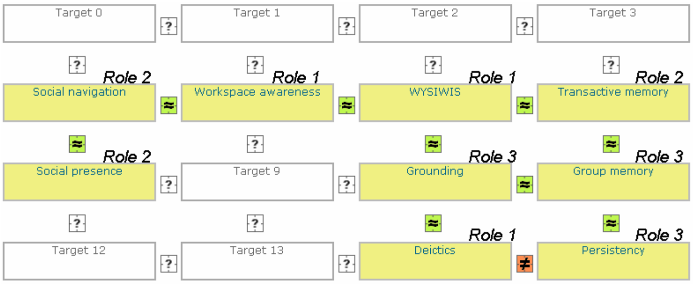
width: 4

height: 4

larger grids (below right) allow students not to place together concepts that have nothing to do with each other. The grids below are annotated with the role played by each cell creator



Role 3 Media space	Role 2 Distributed teams	Role 1 Media richness
Role 2 Informal communication	Role 2 Co-presence	Role 1 Imitation bias
Role 2 Group mirror	Role 3 Turn taking mechanism	Role 3 Gaze awareness



Target 0	Target 1	Target 2	Target 3
Role 2 Social navigation	Role 1 Workspace awareness	Role 1 WYSIWIS	Role 2 Transactive memory
Role 2 Social presence	Target 9	Role 3 Grounding	Role 3 Group memory
Target 12	Target 13	Role 1 Deictics	Role 3 Persistency

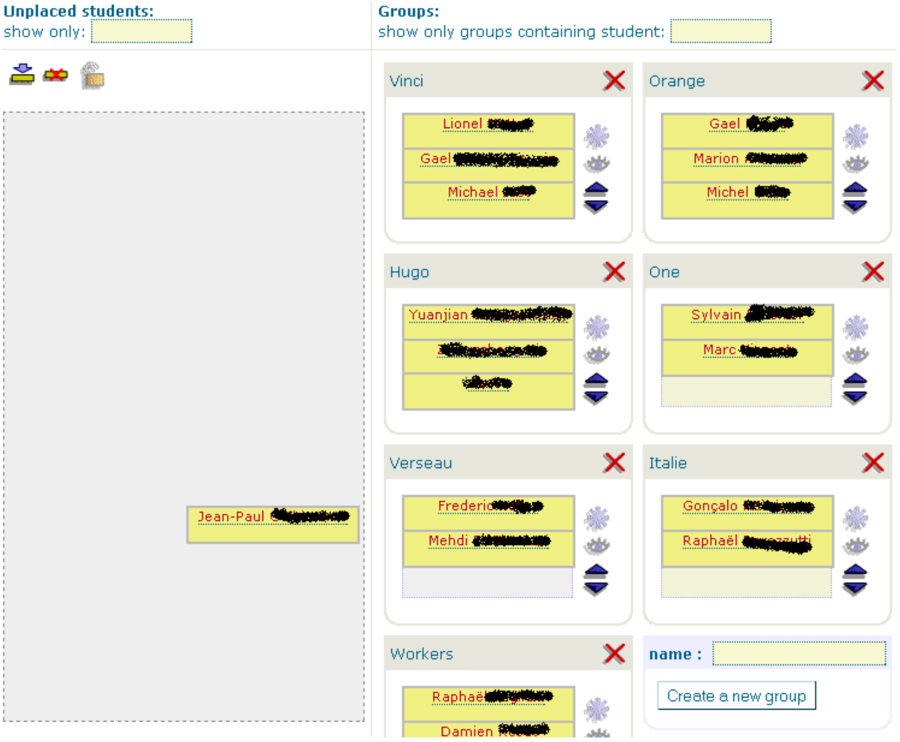
(step 6) How to form groups

Groups may be formed by the students themselves or by the teacher. We recommend to let students form group by themselves. However, the teacher may choose to do it for different reasons: for instance if there are 30 students in biology and 15 students in chemistry, one could make groups of 2 biologists and 1 chemist to increase the spirit of a Jigsaw.

Running Phase 1:

Group formation

This phase is not a learning activity. All registered students appear in the left grey area. Names can be dragged and dropped from this initial space to the groups and vice-versa. Every new group needs to have a name. By default, the size of a group is the number of roles defined.



Unplaced students:

show only:

Jean-Paul [redacted]

Groups:

show only groups containing student:

Vinci: Lionel [redacted], Gael [redacted], Michael [redacted]

Orange: Gael [redacted], Marion [redacted], Michel [redacted]

Hugo: Yuanjian [redacted], [redacted], [redacted]

One: Sylvain [redacted], Marc [redacted]

Verseau: Frederic [redacted], Mehdi [redacted]

Italie: Gonçalo [redacted], Raphaël [redacted]

Workers: Raphaë [redacted], Damien [redacted]

name :

Automatic or manual

The button randomly distributes the students from the left panel to the remaining seats in the groups. This automatic group formation can be combined with manual modifications. It makes sense only for large classes. The reverts the process and brings all students back to the left pane. It is convenient to give names to the groups.

Locking / unlocking

The and buttons enable or disable the possibility for students to form groups by themselves. After phase 1, the groups are automatically locked. Therefore, choosing the right dates for this phase is very important.

The two areas entitled "show only" enable to display only some students. They are useful for filtering long list of names that could not appear in one screen, making 'drag & drop' difficult.

Missing or extra students?

If 16 students have to form teams of 3, what should the 16th student do? If the class has 14 students, one group will not be complete. ManyScripts can cope with these constraints in the following way:

If there are fewer students than roles, ManyScripts suggests the "SPY" feature: if role 3 is missing in team "Italie", this team may borrow the definitions produced by any student playing role 3 in another team.

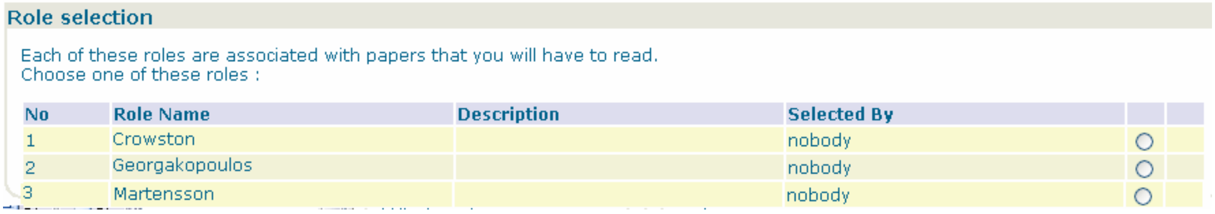
If there are more students than roles, ManyScript suggests the "JOKER" feature: the extra student(s) will have the right to act as any of the other roles of his team. If the team has 9 concepts to define now includes 4 members instead of 3, they will freely decide how to distribute the workload among themselves.

These solutions are not perfect since workload is somewhat uneven in groups but they enable the teacher to continue the script despite unexpected events (e.g. student dropout).

Running Phase 2:

Role Selection

Each student simply chooses the role he decides to play, usually after negotiating with his or her colleagues. The criteria students often use to pick role are not much about personal preferences but rather the amount of work: whether they already know something about this role and the length of the papers to read, etc.



Role selection

Each of these roles are associated with papers that you will have to read.
Choose one of these roles :

No	Role Name	Description	Selected By		
1	Crowston		nobody	○	
2	Georgakopoulos		nobody	○	
3	Martensson		nobody	○	

Running Phase 3:

Students download and usually print the papers and

Reading papers

(hopefully) read them, looking for the concepts that need to be defined.

Running Phase 4:

Distributing concepts

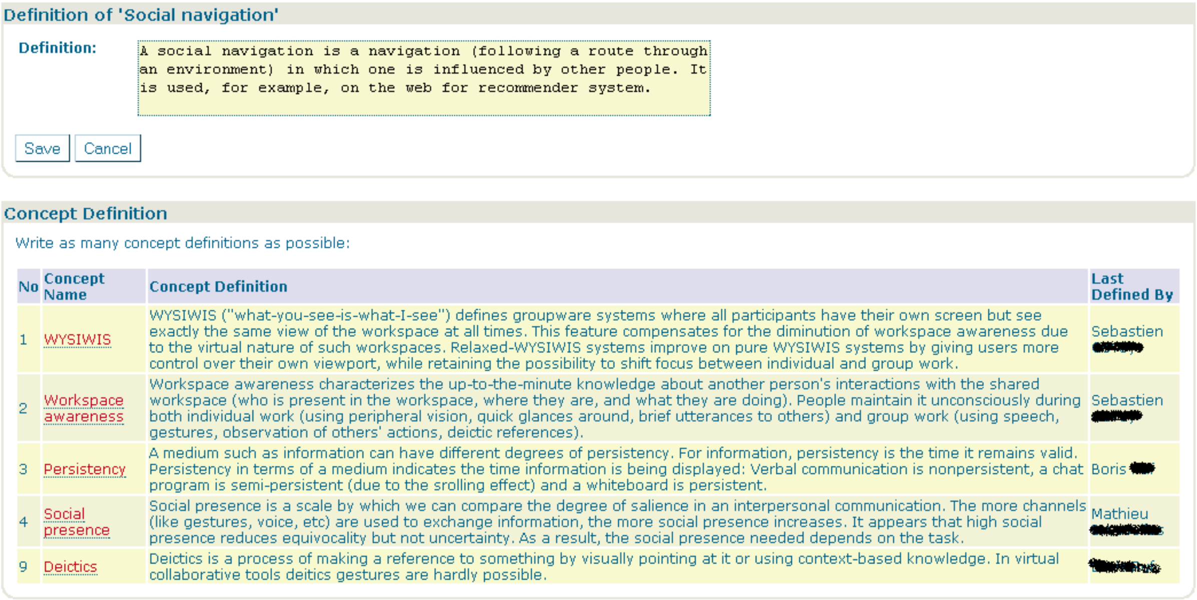
Based on their readings, teams define which member will define which concepts. This phase is not supported in the tool, they simply discuss this face-to-face or via any communication tool. ManyScripts does not know which student will define which concepts. Students are free to change "who defines what" as often as they want.

Running Phase 5:

Defining concepts

Students, usually individually, enter the definition of the concepts. They may define any concept, including revising definitions provided by other students. Experience shows that they sometimes edit minor details of their peer definition, but rarely rephrase it at a conceptual level. It is important to recommend them to save their work on a regular basis.

As shown in the next figure, they can see all the definitions provided and also who has edited the definition for the last time.



Definition of 'Social navigation'

Definition: A social navigation is a navigation (following a route through an environment) in which one is influenced by other people. It is used, for example, on the web for recommender system.

Concept Definition

Write as many concept definitions as possible:

No	Concept Name	Concept Definition	Last Defined By
1	WYSIWIS	WYSIWIS ("what-you-see-is-what-I-see") defines groupware systems where all participants have their own screen but see exactly the same view of the workspace at all times. This feature compensates for the diminution of workspace awareness due to the virtual nature of such workspaces. Relaxed-WYSIWIS systems improve on pure WYSIWIS systems by giving users more control over their own viewport, while retaining the possibility to shift focus between individual and group work.	Sebastien [redacted]
2	Workspace awareness	Workspace awareness characterizes the up-to-the-minute knowledge about another person's interactions with the shared workspace (who is present in the workspace, where they are, and what they are doing). People maintain it unconsciously during both individual work (using peripheral vision, quick glances around, brief utterances to others) and group work (using speech, gestures, observation of others' actions, deictic references).	Sebastien [redacted]
3	Persistency	A medium such as information can have different degrees of persistency. For information, persistency is the time it remains valid. Persistency in terms of a medium indicates the time information is being displayed: Verbal communication is nonpersistent, a chat program is semi-persistent (due to the srolling effect) and a whiteboard is persistent.	Boris [redacted]
4	Social presence	Social presence is a scale by which we can compare the degree of salience in an interpersonal communication. The more channels (like gestures, voice, etc) are used to exchange information, the more social presence increases. It appears that high social presence reduces equivocality but not uncertainty. As a result, the social presence needed depends on the task.	Mathieu [redacted]
9	Deictics	Deictics is a process of making a reference to something by visually pointing at it or using context-based knowledge. In virtual collaborative tools deitics gestures are hardly possible.	[redacted]

Quality of definitions is a concern

The size of the text entry zone indicates to students that they are expected to enter short definitions, not long texts. Experience shows that students, even master students, are not very good in producing definitions.

- Some students copy definition from the papers they read. Sometimes they cut and paste 20 lines, expecting the teacher will find something interesting in it!
- Other students simply select any sentence from the paper that mentions the concept without being a real definition. For instance, instead of defining wine as "an alcoholic beverage made from grape fermentation", they will type something like "During wedding ceremonies, many guests drink wine".
- Students tend to confuse a definition with a property, for instance "wine is red, rosé or white".

- Students tend to pick under-generalized definition such as "knowledge management aims at storing employee's knowledge in a database" although there are many approaches to knowledge management.

Discouraging 'cut & paste'

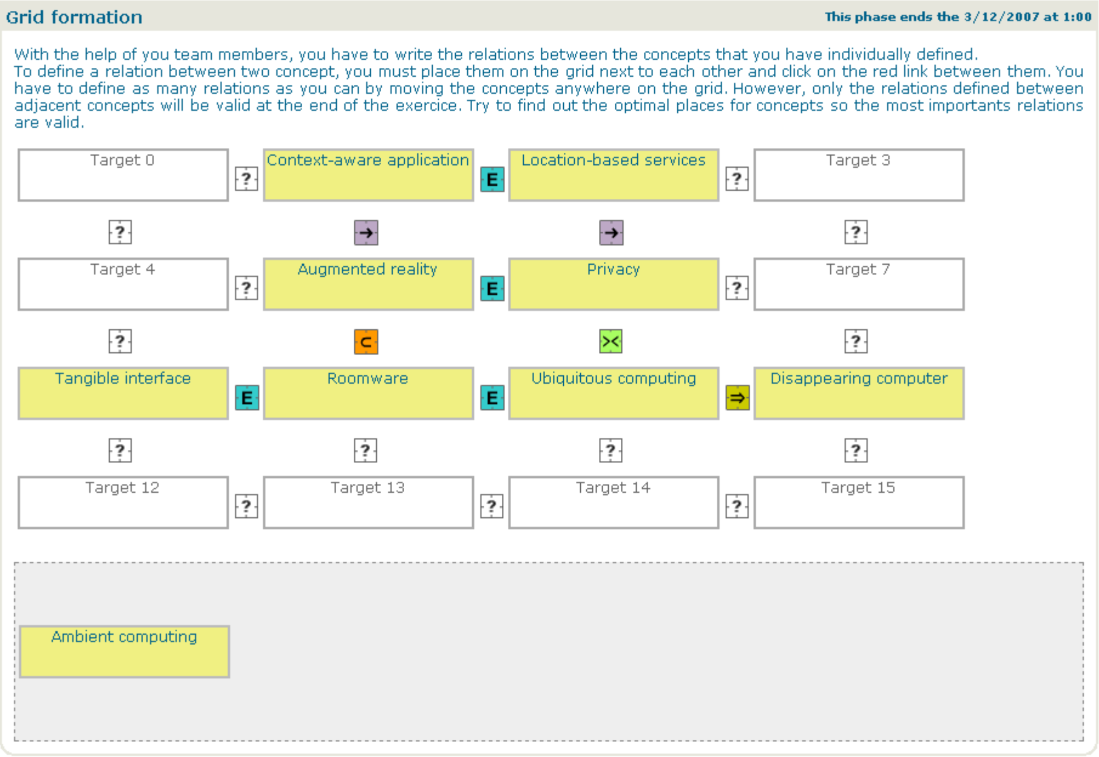
One way to reduce this "cut and paste " habit is to ask students to write the definitions in a different language than the one used in the papers. Typically, we had better definitions from students who built grids in French when the papers they had to read were in English. The second way to reduce "cut and paste" is to ask students during the debriefing session, to explain the definitions they have entered in their own words.

Running Phase 6:

Grid Construction

Students have to place the concepts on the grid in such a way that they are able to define the relationship between pairs of juxtaposed concepts. Please be aware that the concepts to be dragged and dropped are located below the grid.

The students usually sit together in front of a computer for this phase.



Grid formation

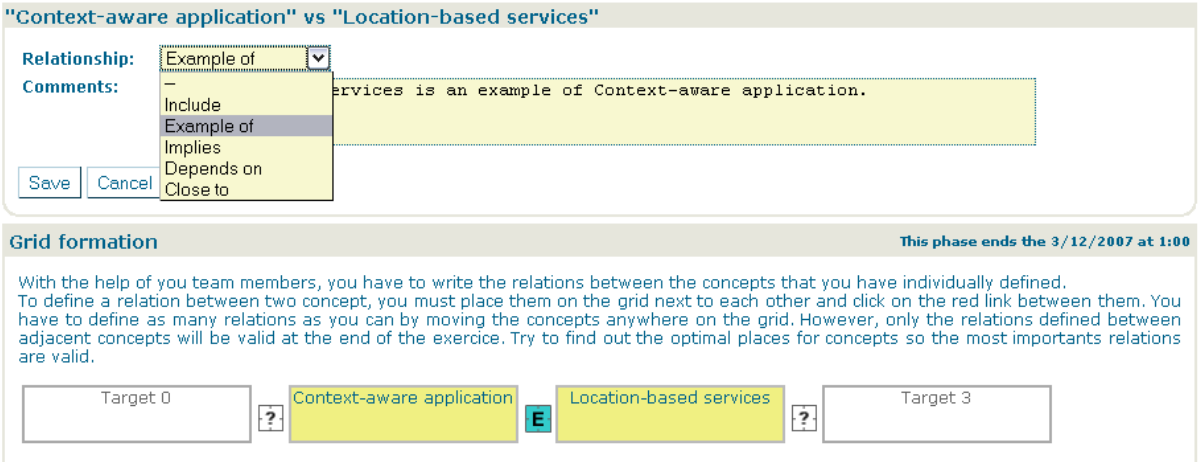
This phase ends the 3/12/2007 at 1:00

With the help of you team members, you have to write the relations between the concepts that you have individually defined.
To define a relation between two concept, you must place them on the grid next to each other and click on the red link between them. You have to define as many relations as you can by moving the concepts anywhere on the grid. However, only the relations defined between adjacent concepts will be valid at the end of the exercice. Try to find out the optimal places for concepts so the most importants relations are valid.

Target 0	?	Context-aware application	E	Location-based services	?	Target 3
?		→		→		?
Target 4	?	Augmented reality	E	Privacy	?	Target 7
?		C		><		?
Tangible interface	E	Roomware	E	Ubiquitous computing	⇒	Disappearing computer
?		?		?		?
Target 12	?	Target 13	?	Target 14	?	Target 15

Ambient computing

In the snapshot below, the concepts "context-aware application" and "location-based services" have been juxtaposed. The student must define the relation between these 2 concepts by entering a short text and selecting a relationship type. ManyScripts proposes 5 types of relationship that are depicted on the grid with different icons.



"Context-aware application" vs "Location-based services"

Relationship: Example of

Comments: ervices is an example of Context-aware application.

Grid formation

This phase ends the 3/12/2007 at 1:00

With the help of you team members, you have to write the relations between the concepts that you have individually defined.
To define a relation between two concept, you must place them on the grid next to each other and click on the red link between them. You have to define as many relations as you can by moving the concepts anywhere on the grid. However, only the relations defined between adjacent concepts will be valid at the end of the exercise. Try to find out the optimal places for concepts so the most importants relations are valid.

Types of relationship between concepts A and B are depicted by the following icons:

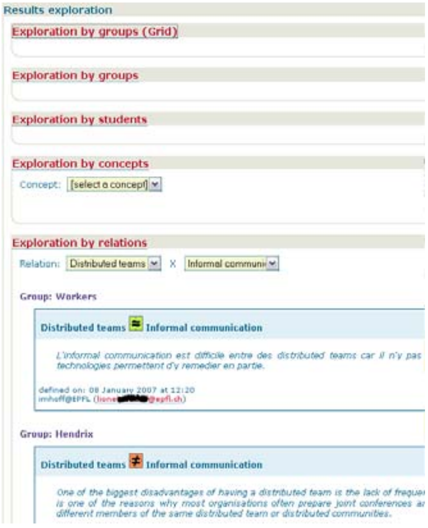
⊂	"Includes" (A is part of B, A is included in B)
E	"Example of" (A is an example of B, A belongs to set B)
⇒	"Implies" (A is the cause of B, B results from A)
→	"depends on" (A depends on B, A uses B, B is necessary to A)
><	"close to" (A and B are similar but should not be confused)

Running Phase 7:

Debriefing

The goal of this phase is to give feedback with regards to definitions and grids and to provide additional information. All students have to be in the classroom.

The teacher cockpit enables exploring the students' productions grid-by-grid, by groups, by students, by concepts or by relations between concepts. This exploration can be done in real time during the debriefing session. However, we usually do it before the course. Grids should be completed the day before the course, so that the teachers would have the evening to explore definitions and find the cases that are most interesting to comment upon or to ask for comments.



Results exploration

Exploration by groups (Grid)

Exploration by groups

Exploration by students

Exploration by concepts

Concept: [select a concept]

Exploration by relations

Relation: Distributed teams X Informal communi

Group: Workers

Distributed teams ⇒ Informal communication

L'informal communication est difficile entre des distributed teams car il n'y pas technologies permettant d'y remedier en partie.

defined on: 08 January 2007 at 12:20
imhoff@EPFL (lione[redacted]@epfl.ch)

Group: Hendrix

Distributed teams ≠ Informal communication

One of the biggest disadvantages of having a distributed team is the lack of frequei is one of the reasons why most organisations often prepare joint conferences ar different members of the same distributed team or distributed communities.

Students feel challenged by viewing their own product publicly displayed by the beamer.

The teacher may ask students :

- to explain their definitions, namely to check if they have understood what they typed;
- to find the difference between two definitions of the same concept;
- to find clusters of related concepts that appear on most grids;
- to find conflicts between different grids.

Trial and error

To let students explore different grid configurations freely, the grid has a "memory". For instance, if the concept "deictics" is now moved to cell 15, its relations with "social presence" will disappear from the screen. However, it is saved in memory and if the students place "deictics" again next to "social presence", their relationship will be displayed again.

Concept Gird without computers ?

ConceptGrid can be run without computers. Teams could write concept definitions on small pieces of paper that could then be assembled on a large piece of a paper (e.g. on a flipchart). Before the lesson, all flipcharts would be displayed around the classroom. The teacher and the students will make a tour of all posters and discuss the definitions. This technology free script offers several advantages (flexibility, global view) but does not benefit from the logistics of ManyScripts (e.g. gathering all definitions for the same concept).

5. The IceGrain script

Overview of the script

IceGrain and IceCube are two scripts based on "peer review". The ICE idea has been developed by the Swiss Centre for Innovation in Learning (scil). Students are expected to learn by criticizing each other's work. This script can be applied to small products, for instance, when each student has to collect 20 images, or to larger products, for instance if each student has to produce an essay. The activities are different in these two cases and hence, instead of having scripts with many parameters to tune, we produced two different scripts. IceGrain is designed for situations where students have to provide multiple small contributions

Phase 1

The students produce a certain number of contributions. Contribution include quiz answers, free text and pictures.

Phase 2

Each student gives feedback to another student's contribution.

Phase 3

Each student revises the feedback that he or she gave by comparing it to the others' feedback.

Phase 4

Each student revises his or her initial contribution based on the feedback he or she received.

Phase 5

The teacher discusses all contributions and feedback in order to let them find emerging evaluation criteria or revise the initial criteria based on their experience.

Choosing an IceGrain script:

Learning objectives

IceGrain can be used in classes where students have to produce contributions that cannot be assessed with a simple or single criterion but require a more subtle judgement. The knowledge necessary to make this judgment is the content of the course; thus, in comparison to ConceptGrid and ArgueGraph, this script aims at procedural knowledge.

Scope

It would be a mistake to believe that IceGrain only applies to domains where personal opinions matters. The contributions elaborated or collected by the student may also be exploited for scientific courses. Teachers may, for instance, ask for the following contributions:

- in medical training, a collection of X-rays pictures that illustrate a pathology;
- in physics, a collection of graphs produced with MatLab;
- in geometry, a collection of geometrical figures that prove or discard a principle;
- in biology, a collection of proteins that have specific properties;
- in literature, a collection of poem excerpts that illustrate surrealist ideas;
- in education, didactic concepts for a certain course.

Inductive or deductive

The students have to collect or to produce a large set of contributions that can be used in two ways.

- In an inductive approach, the set of contributions will be used for constructing general principles (definitions, rules, ontologies,…) and articulating them in a theory. This construction will be conducted during the debriefing phase. The teacher will exploit the mutual critiques as the raw material from which general principles may be abstracted.
- In a deductive approach, the general principles should be presented in an introductory lecture before starting the script. The students are then expected to apply these principles when elaborating their contributions as well as to refer to these principles when commenting on their peers' products. In the debriefing phase, students will be asked to discuss how the general principles have been applied and how they may be revised after the peer review experience.

What should be chosen?

The inductive approach implements the socio-constructivist principles according to which students will better understand, remember and apply the principles if they are grounded in a meaningful social experience. However, it is more difficult to put into action as the debriefing phase requires some improvisation from the teacher.

Deduction is easier because the principles to be introduced are closely related to the perceptual features of the instances collected. In the debriefing phase, the teacher will ask students to find features shared by the positive instances and absent from the negative instances. He or she will focus on near-miss instances, i.e. objects that have all features of the concept but one and hence reveal the importance of this feature for the concept. In many domains, the general principles have been elaborated through centuries of research and one cannot expect students to induce them with a few examples or within a few hours. In this case, the deductive approach is recommended.

Class size

The IceCube can be applied to large classes since only the debriefing phase requires co-presence. The deductive approach is easier to apply to large classes because the debriefing phase is less difficult to conduct. It can probably be applied, for instance, to classes of 200 or 300 students, however, we have not tested this yet. The feedback assignment should be done automatically (because the manual assignment window would then be a 300 X 300 matrix).

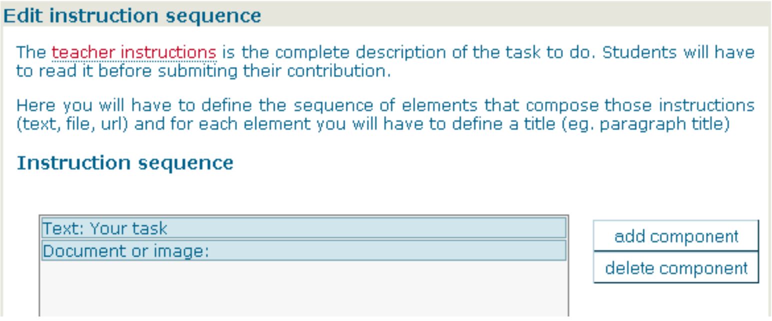
Online implementation

The deductive approach can be used completely online since the debriefing phase is less important or could be done via a virtual classroom.

Creating the script contents:

(step 1) Writing instructions for students

The first step is to define the instructions that inform students which products they are expected to upload in phase 1. The item "define task instructions" includes a very basic page editor. First, the teacher defines the structure of the page which can include text, pictures and URLs.



Edit instruction sequence

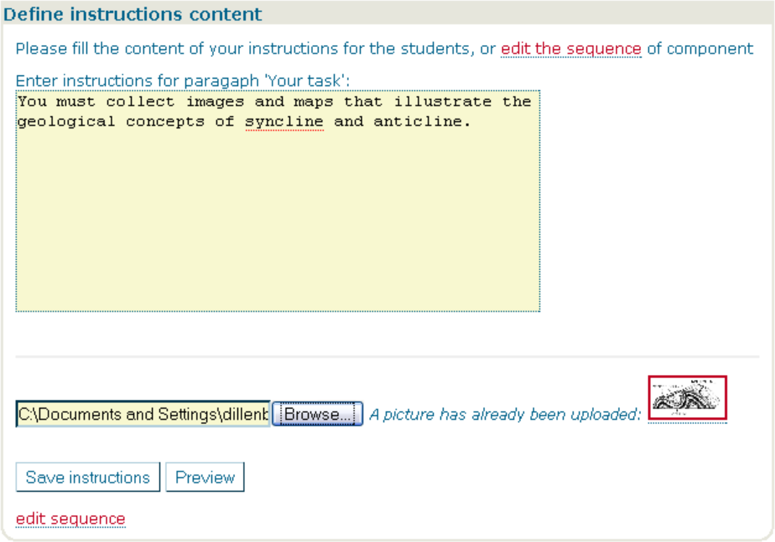
The teacher instructions is the complete description of the task to do. Students will have to read it before submiting their contribution.

Here you will have to define the sequence of elements that compose those instructions (text, file, url) and for each element you will have to define a title (eg. paragraph title)

Instruction sequence

Text: Your task
Document or image:

Then, the teacher has to fill the page template defined in the previous step. In most cases, instructions are a short text. In the example below, the teacher defines a layout that may include a text, a picture or any document that student could download. Then, the link "instructions content page" opens the pane below in which the teacher defines the content of the components previously created.



Define instructions content

Please fill the content of your instructions for the students, or edit the sequence of component

Enter instructions for paragaph 'Your task':

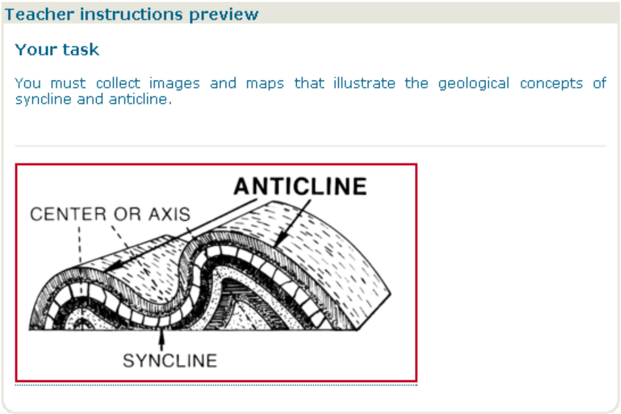
```
You must collect images and maps that illustrate the
geological concepts of syncline and anticline.
```

C:\Documents and Settings\dillenk

A picture has already been uploaded:

Preview

The button "preview" displays the instruction as students will see them. You can return to the previous step with the link "edit sequence".



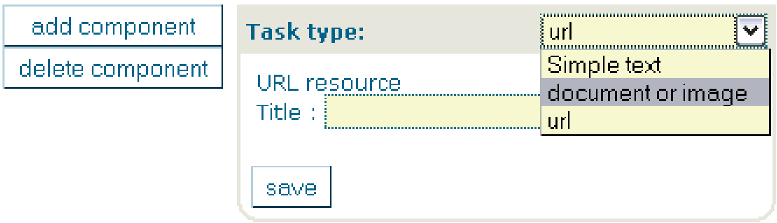
Teacher instructions preview

Your task

You must collect images and maps that illustrate the geological concepts of syncline and anticline.

For complex instructions, use an editor

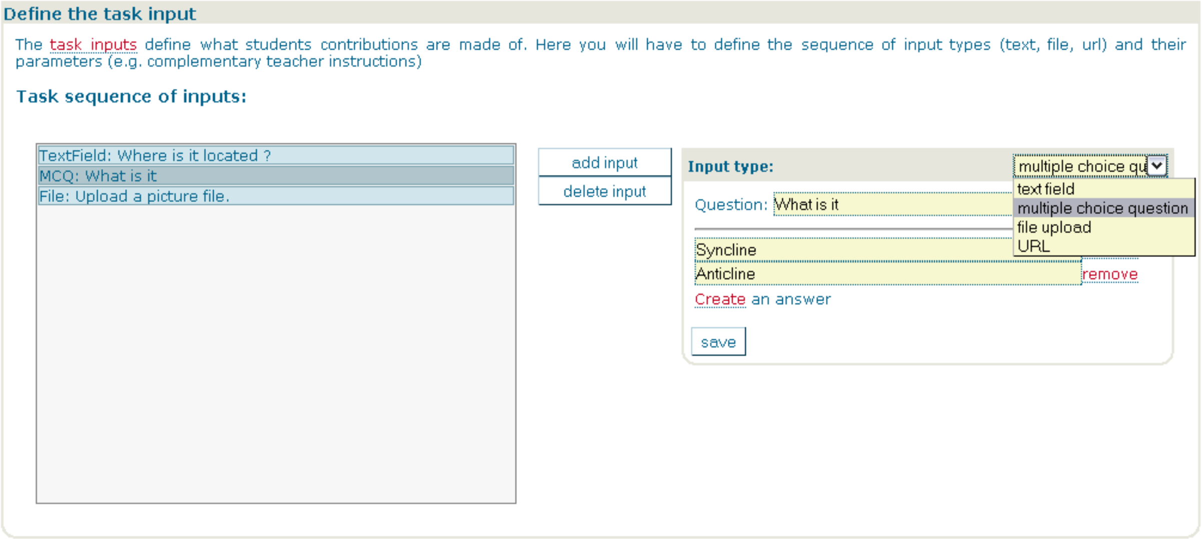
We did implement a full page editor: the system only supports plain text and pictures. For instructions requiring a more sophisticated page layout, the author should use any other software and import the results as a document and the URL. This import is defined when adding a component (left button) in when pre-defining the instructions page.



(Step 2) Defining contributions

The author has to define what students will be asked to enter in the environment: "Define task input". The teacher may ask students to enter text, to select an option in a multiple choice questionnaire, to upload a file or to enter an URL.

Typically, if students have to enter an image, this can be done by uploading the image file or by giving the URL of a web site. In the example below, students will have to enter the location of the geological picture as a text, to specify if it is a syncline or anticline and to upload a picture as a file. Every student's contribution will need to include these elements.



Define the task input

The task inputs define what students contributions are made of. Here you will have to define the sequence of input types (text, file, url) and their parameters (e.g. complementary teacher instructions)

Task sequence of inputs:

TextField: Where is it located ?
MCQ: What is it
File: Upload a picture file.

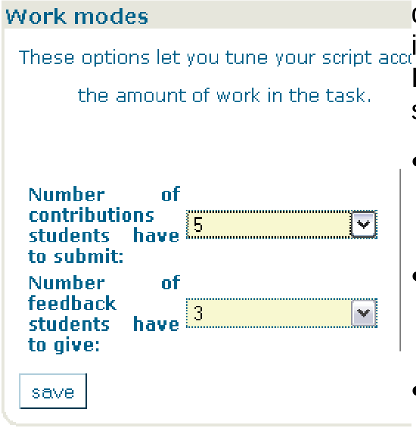
Input type: multiple choice qu

Question: What is it

Syncline
Anticline

(Step 3) Defining the number of contributions

Finally, the author has to define the number of inputs. In the example below, each student has to enter 5 contributions, which means 5 times the 3 inputs specified in the previous step (location's name, type, picture file). The author also has to decide how many contributions each student has to comment in phase 2. The number of feedbacks to provide is important. Let's imagine that each student produces 5 examples of syncline/anticline. Different situations can be created:



Work modes

These options let you tune your script acc

the amount of work in the task.

Number of contributions students have to submit: 5

Number of feedback students have to give: 3

- If each student is asked to comment on 3 examples produced by other students, many examples (2/5) will not receive any feedback.
- If each student is asked to comment on 5 examples produced by other students, all examples will receive a single feedback which is not very convincing.
- If each student is asked to comment on 10 examples produced by other students, all examples will receive 2 feedbacks.

Defining the session

(Step 4) Set up the timing

(Step 5) Changing parameters

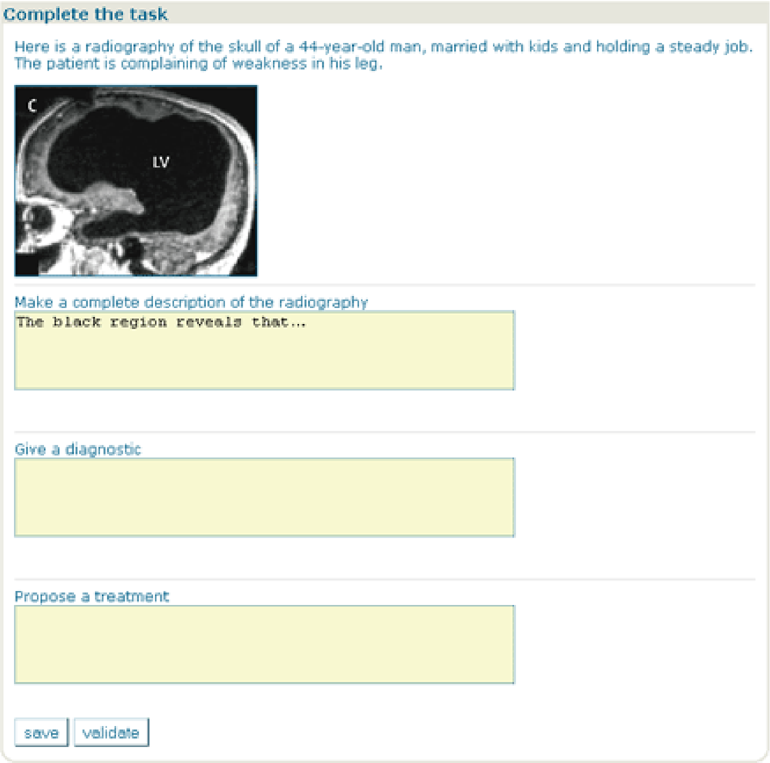
When creating the session, the teacher has to enter the dates of each phase. The two parameters "number of contributions" and "number of feedback" can also be modified at the session level.

Running Phase 1:

Individual contribution

The script starts with a short session where the teacher explains the expectations for the different phases. Then, during phase 1, students have to enter as many contributions as defined by the teacher within the time frame.

Students enter their contributions



Complete the task

Here is a radiography of the skull of a 44-year-old man, married with kids and holding a steady job. The patient is complaining of weakness in his leg.

Make a complete description of the radiography

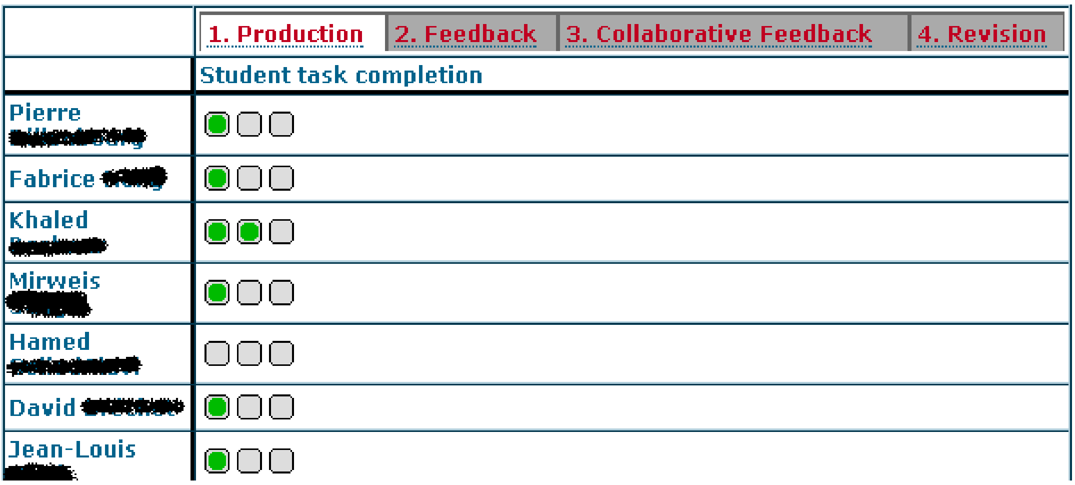
The black region reveals that...

Give a diagnostic

Propose a treatment

Teacher monitors students' work

The teacher can see (menu 'cockpit', option 'follow up') how far the students have proceeded in their work. The window below is organized as 4 thumbnails. The first thumbnail ("Production") show that most students have so far made 1 contributions but that Khaled already completed 2 and Hamed zero. Clicking on green dots that represent a contribution displays a trace of this contribution across all phases of the script. The number of dots represents the number of expected contributions. If the dot is only partly colored in green, it means the students has only provided a subset of the elements of this contribution.



	1. Production 2. Feedback 3. Collaborative Feedback 4. Revision
	Student task completion
Pierre [redacted]	🟩 □ □
Fabrice [redacted]	🟩 □ □
Khaled [redacted]	🟩 🟩 □
Mirweis [redacted]	🟩 □ □
Hamed [redacted]	□ □ □
David [redacted]	🟩 □ □
Jean-Louis [redacted]	🟩 □ □

This tracing tool enables the teacher to anticipate that some students will be late and send them some warning, to have a glimpse on their contributions and repeat or revise instructions if these productions do not correspond to his expectations. As with any pedagogical method, what happens during this script needs to be permanently monitored and adjusted if necessary

Running Phase 2:

Feedback

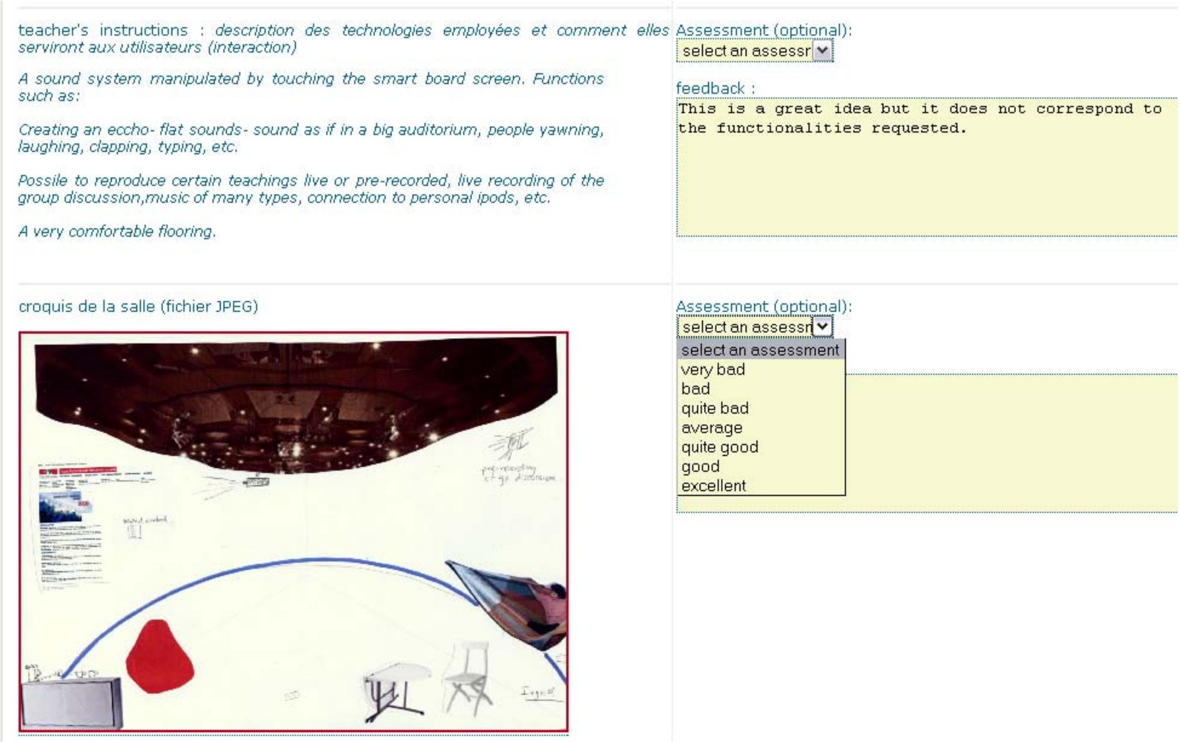
Feedback is requested for each component of the contribution: a student may provide a positive feedback for a picture but a negative feedback for the title given to this picture.

Each feedback includes free text and a judgment on a scale from "very bad" to "very good". This quantitative feedback may be especially useful to identify very good or very bad contributions when the script session involves many students or many contributions per students.

Useful feedback

The students must be explicitly told to provide feedback that is not only polite, but also tells something about how to revise the contribution later on. We suggest to give them instructions such as:

"Please express your feedback in a polite way. Be critical about the contributions of your colleague but not destructive. The best feedback is the one that helps your colleague to improve his or her contribution later on. Please refer to the course concepts when writing your comments"



teacher's instructions : *description des technologies employées et comment elles serviront aux utilisateurs (interaction)*

A sound system manipulated by touching the smart board screen. Functions such as:

Creating an eccho- flat sounds- sound as if in a big auditorium, people yawning, laughing, clapping, typing, etc.

Possile to reproduce certain teachings live or pre-recorded, live recording of the group discussion,music of many types, connection to personal ipods, etc.

A very comfortable flooring.

Assessment (optional):

select an assessr

feedback :

This is a great idea but it does not correspond to the functionalities requested.

croquis de la salle (fichier JPEG)

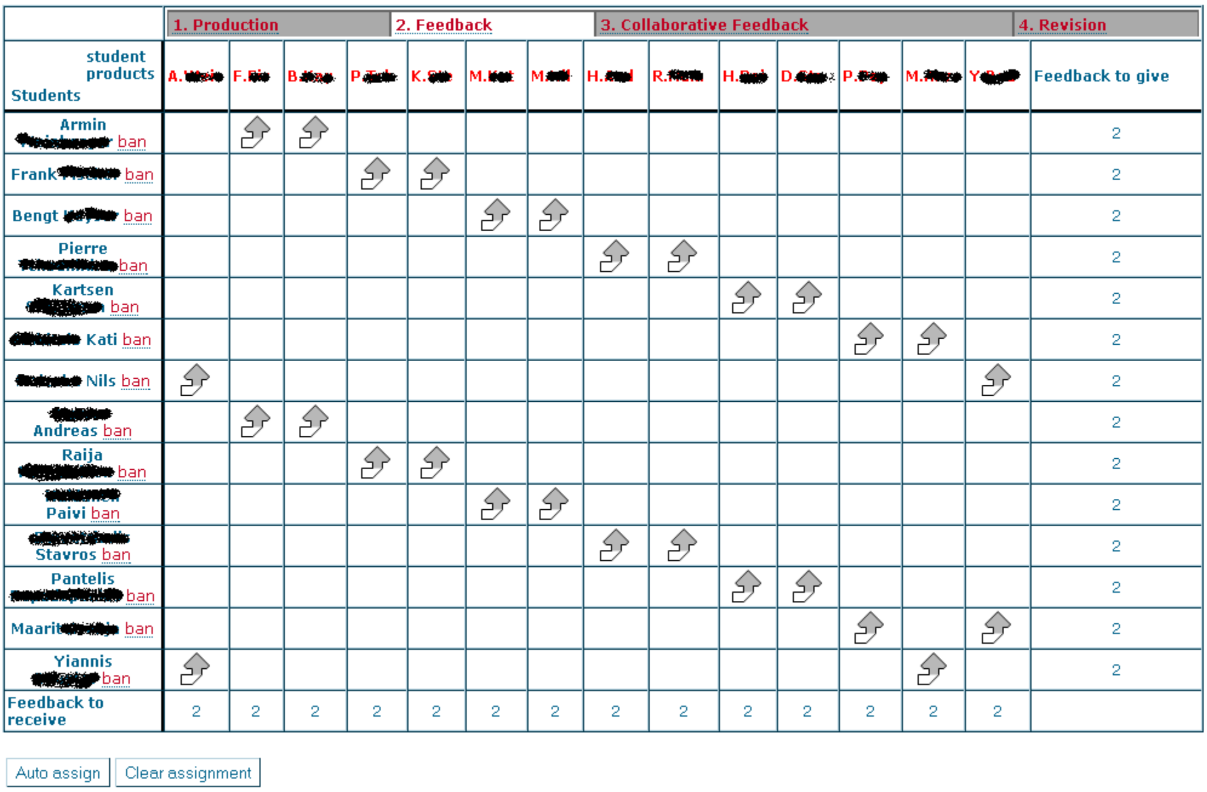
Assessment (optional):

select an assessn

Who gives feedback to whom?

The teacher decides who gives feedback to whom by using the window below. In the 'cockpit' menu (option 'follow-up'), the thumbnail 'feedback' displays a matrix of which students (in rows) will feedback the contributions of which students (in columns). The arrows in a cell (*i j*) indicate that the student in the row *i* will review the contribution of the student in column *j*. The sums of rows and the sums of columns enable the teacher to equilibrate the workload. However, a teacher may decide that some students get more or get less to do. In case of automatic assignment, the sum of rows and columns will match the parameters defined by the teacher, but in case of manual assignment, they may be different.

We recommend a mixed assignment method: first, the teacher may use the button 'auto assign' to distribute automatically the assignments as illustrated below; second, some assignments can be manually changed by a right-click on the arrow icon in the cell to be modified. Such changes might for instance take into consideration that two students hate each other or have been working too closely together. In case of manual changes, the teacher has to check if the sums of rows and the sums of columns nonetheless match the parameters.



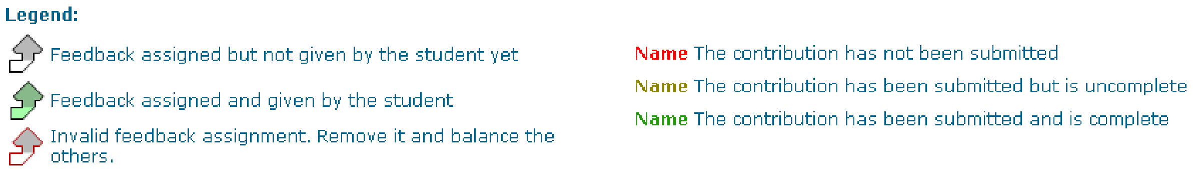
student products Students	A.[redacted]	F.[redacted]	B.[redacted]	P.[redacted]	K.[redacted]	M.[redacted]	M.[redacted]	H.[redacted]	R.[redacted]	H.[redacted]	D.[redacted]	P.[redacted]	M.[redacted]	Y.[redacted]	Feedback to give
Armin [redacted] ban		↗	↗												2
Frank [redacted] ban				↗	↗										2
Bengt [redacted] ban						↗	↗								2
Pierre [redacted] ban								↗	↗						2
Kartsen [redacted] ban										↗	↗				2
[redacted] Kati ban												↗	↗		2
[redacted] Nils ban	↗													↗	2
[redacted] Andreas ban		↗	↗												2
Raija [redacted] ban				↗	↗										2
[redacted] Paivi ban						↗	↗								2
[redacted] Stavros ban								↗	↗						2
Pantelis [redacted] ban										↗	↗				2
Maarit [redacted] ban												↗		↗	2
Yiannis [redacted] ban	↗												↗		2
Feedback to receive	2	2	2	2	2	2	2	2	2	2	2	2	2	2	

Teacher monitors students' work

This feedback matrix is used:

- for assigning feedback;
- for monitoring progress, e.g. sending reminders for missing feedbacks);
- for modifying assignments for any reason, such as students dropping out or failing to deliver their contributions in due time.

The colour of arrows indicates the status of the feedback that A (in row) gives to B (column). The legend is indicated below.



Legend:

⇧ Feedback assigned but not given by the student yet

🟩 Feedback assigned and given by the student

🟥 Invalid feedback assignment. Remove it and balance the others.

Name The contribution has not been submitted

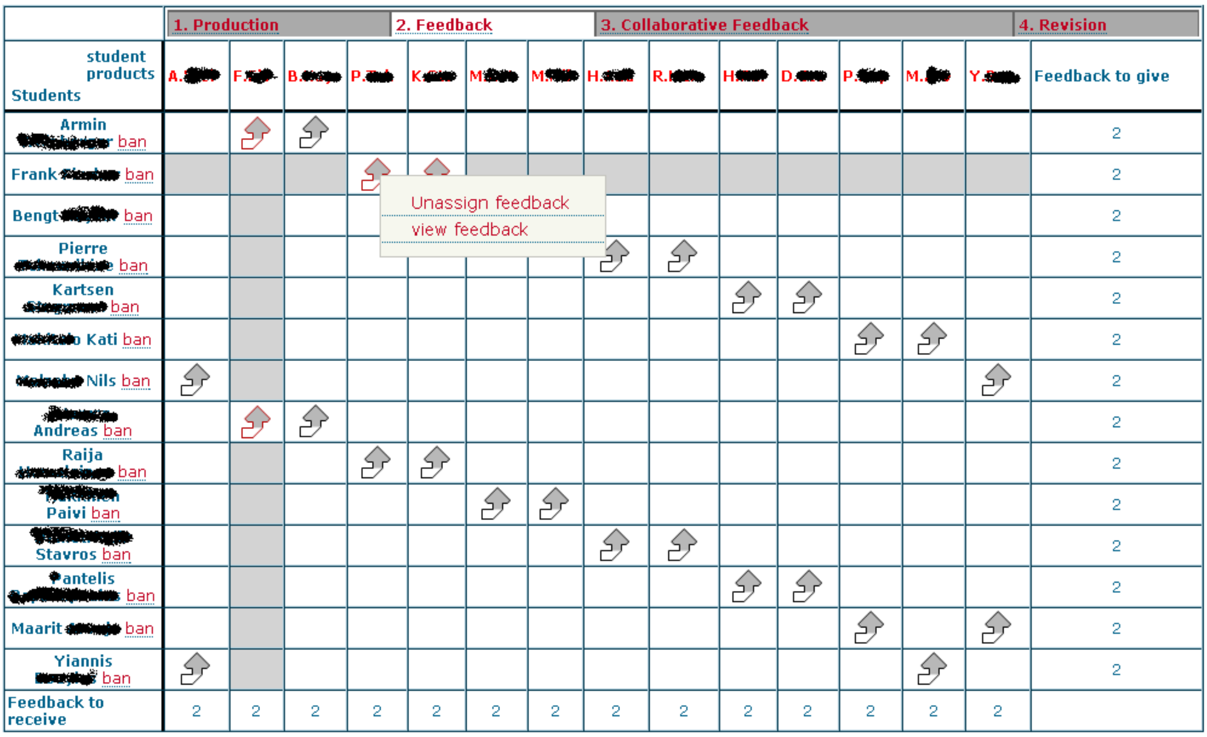
Name The contribution has been submitted but is uncomplete

Name The contribution has been submitted and is complete

Students dropping out

If a student drops out, the teacher will indicate it by selecting the "ban" button next to the student's name. His line and column are then shaded and feedbacks in this row and this column appear in red to indicate that they need to be re-assigned.

Using this feedback matrix seems somewhat complex but it provides a lot of flexibility: manual modifications enable teachers to cope with unexpected events. Whatever happens, the script should proceed.



1. Production | 2. Feedback | 3. Collaborative Feedback | 4. Revision

student products Students	A.[redacted]	F.[redacted]	B.[redacted]	P.[redacted]	K.[redacted]	M.[redacted]	M.[redacted]	H.[redacted]	R.[redacted]	H.[redacted]	D.[redacted]	P.[redacted]	M.[redacted]	Y.[redacted]	Feedback to give
Armin [redacted] ban		🟥	⇧												2
Frank [redacted] ban				🟥	🟥										2
Bengt [redacted] ban															2
Pierre [redacted] ban								⇧	⇧						2
Kartsen [redacted] ban										⇧	⇧				2
[redacted] Kati ban												⇧	⇧		2
[redacted] Nils ban	⇧													⇧	2
[redacted] Andreas ban		🟥	⇧												2
Raija [redacted] ban				⇧	⇧										2
[redacted] Paivi ban						⇧	⇧								2
[redacted] Stavros ban								⇧	⇧						2
[redacted]antelis [redacted] ban										⇧	⇧				2
Maarit [redacted] ban												⇧		⇧	2
Yiannis [redacted] ban	⇧												⇧		2
Feedback to receive	2	2	2	2	2	2	2	2	2	2	2	2	2	2	

Blindness

The students do not see the contributions that have not been assigned to them. Some students asked us to add this functionality. However, experience showed that, if they cannot see the work of others, the debriefing session gets much richer because students are curious to discover the contributions they did not see before.

Anonymous

Should feedback be anonymous or not? Anonymous feedback may help some students to be more frank in their feedback but it may also push some students to be rude. It all depends on the class spirit. This is an option that can be chosen for the session or for the script.

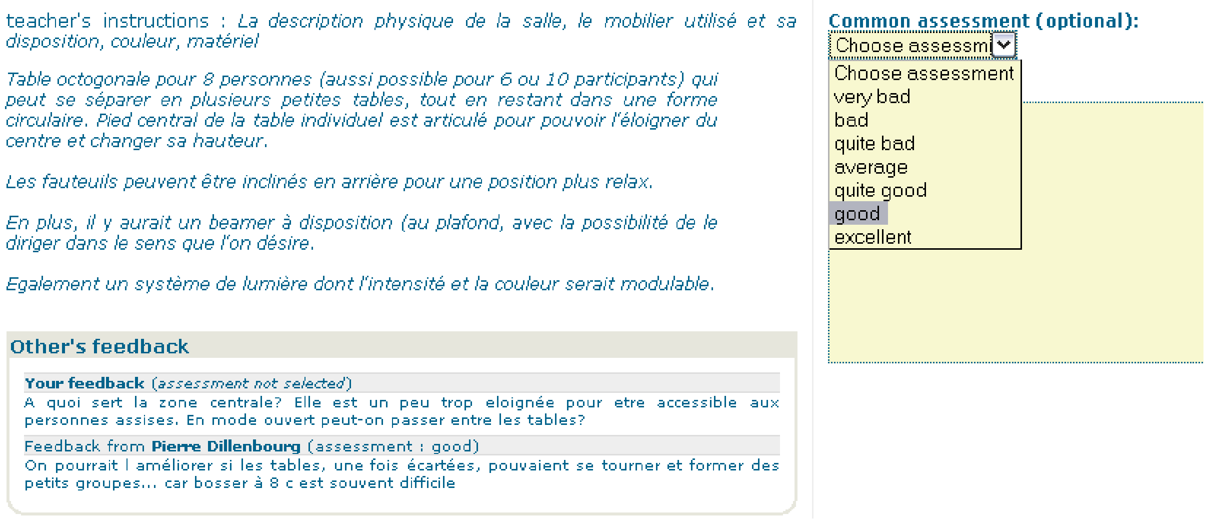
Feedback are anonymous ☑

Running Phase 3:

Feedback revision

In Phase 3, the students have the possibility to compare their feedback to the feedback provided by others and, if they want, to revise their feedback. This is an important phase where they may become aware not only of personal differences of scale when assessing each other's work but, overall, of some criteria that they did not consider. We suggest to provide students with instructions such as:

Please read carefully the feedback provided by others. It may give you ideas for revising your feedback. You do not necessarily have to agree with the opinions of others, but you might pay attention to the criteria that they use.



teacher's instructions : *La description physique de la salle, le mobilier utilisé et sa disposition, couleur, matériel*

Table octogonale pour 8 personnes (aussi possible pour 6 ou 10 participants) qui peut se séparer en plusieurs petites tables, tout en restant dans une forme circulaire. Pied central de la table individuel est articulé pour pouvoir l'éloigner du centre et changer sa hauteur.

Les fauteuils peuvent être inclinés en arrière pour une position plus relax.

En plus, il y aurait un beamer à disposition (au plafond, avec la possibilité de le diriger dans le sens que l'on désire.

Egalement un système de lumière dont l'intensité et la couleur serait modulable.

Other's feedback

Your feedback (*assessment not selected*)
A quoi sert la zone centrale? Elle est un peu trop eloignée pour etre accessible aux personnes assises. En mode ouvert peut-on passer entre les tables?

Feedback from **Pierre Dillenbourg** (assessment : good)
On pourrait l améliorer si les tables, une fois écartées, pouvaient se tourner et former des petits groupes... car bosser à 8 c est souvent difficile

Common assessment (optional):

Running Phase 4:

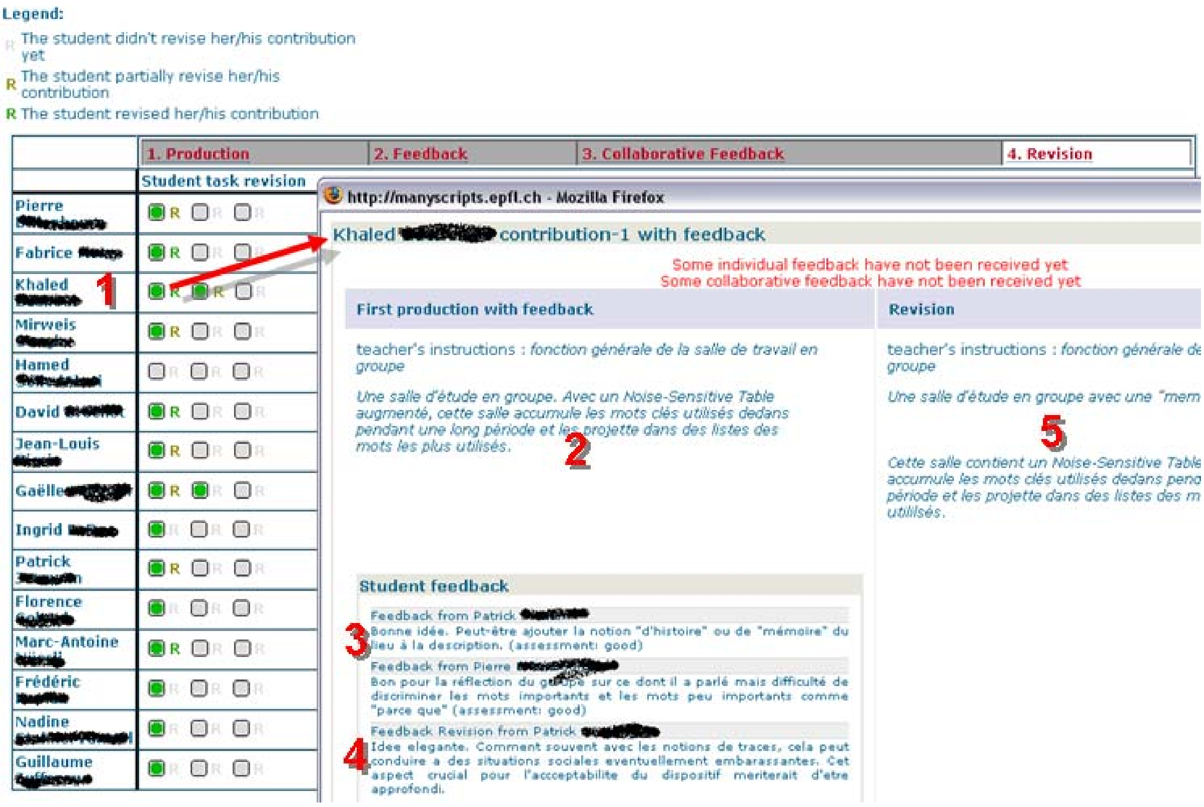
Contribution revision

In this phase, the students have to modify their initial contribution in a way that takes into account (some of) the feedback provided by their peers. Of course, students will often disagree with some of these comments. The following instructions may be given to students at this point:

"Please take into consideration the comments that have been provided to you even if you do not fully agree with them. Please record the comments that you did not take into consideration, either because you disagree with the point or because you did not know how to apply it to your contribution. You will have to explain these points in the final sessions. "

The teacher monitors students' work

As for the previous phases, the follow-up window enables the teacher to see which contributions have been partly or fully revised. By clicking on one of the contribution icons (point 1 in the snapshot below), the teacher opens a window that display Khaled's initial contribution (2), the feedback provided by Pierre and Patrick (3), Patrick's revised feedback (4) and Khaled's revised contribution.



Legend:

R The student didn't revise her/his contribution yet

R The student partially revise her/his contribution

R The student revised her/his contribution

	1. Production	2. Feedback	3. Collaborative Feedback	4. Revision
	Student task revision			
Pierre [redacted]	🟩 R □ R □ R			
Fabrice [redacted]	🟩 R □ R □ R			
Khaled [redacted]	🟩 R 🟩 R □ R			
Mirweis [redacted]	🟩 R □ R □ R			
Hamed [redacted]	□ R □ R □ R			
David [redacted]	🟩 R □ R □ R			
Jean-Louis [redacted]	🟩 R □ R □ R			
Gaëlle [redacted]	🟩 R 🟩 R □ R			
Ingrid [redacted]	🟩 R □ R □ R			
Patrick [redacted]	🟩 R □ R □ R			
Florence [redacted]	🟩 R □ R □ R			
Marc-Antoine [redacted]	🟩 R □ R □ R			
Frédéric [redacted]	🟩 R □ R □ R			
Nadine [redacted]	🟩 R □ R □ R			
Guillaume [redacted]	🟩 R □ R □ R			

http://manyscripts.epfl.ch - Mozilla Firefox

Khaled [redacted] contribution-1 with feedback

Some individual feedback have not been received yet
Some collaborative feedback have not been received yet

First production with feedback	Revision
teacher's instructions : *fonction générale de la salle de travail en groupe*	teacher's instructions : *fonction générale de groupe*
Une salle d'étude en groupe. Avec un Noise-Sensitive Table augmenté, cette salle accumule les mots clés utilisés dedans pendant une long période et les projette dans des listes des mots les plus utilisés.	*Une salle d'étude en groupe avec une "mem* *Cette salle contient un Noise-Sensitive Table accumule les mots clés utilisés dedans perio période et les projette dans des listes des m utililsés.*

Student feedback

Feedback from Patrick [redacted]
Bonne idée. Peut-être ajouter la notion "d'histoire" ou de "mémoire" du lieu à la description. (assessment: good)

Feedback from Pierre [redacted]
Bon pour la réflection du groupe sur ce dont il a parlé mais difficulté de discriminer les mots importants et les mots peu importants comme "parce que" (assessment: good)

Feedback Revision from Patrick [redacted]
Idee elegante. Comment souvent avec les notions de traces, cela peut conduire a des situations sociales eventuellement embarassantes. Cet aspect crucial pour l'accceptabilite du dispositif meriterait d'etre approfondi.

Running Phase 5:

Debriefing

The debriefing phase is based on the follow-up tool previously described. There are several non-exclusive ways for the teacher to exploit the contributions during the debriefing:

- With small classes, the teacher may ask some/all students to present to the whole class their revised contributions and to explain what they learned from their peers' feedback.
- The teacher may analyse contributions before the debriefing phase and select those that might lead to more interesting discussions, namely because the revision process illustrates the key elements that the teacher wanted to address in the course.
- The teacher may analyse contributions before the debriefing phase and organize them into categories in such a way that a set of categories reflects the ontology that he or she intended to teach in this course.
- The teacher may collect all criteria that have been used in feedback, analyse which ones had a significant impact on revision, and classify these criteria per categories, per theory, ….

6. The IceCube script

Overview of the script

The IceCube script was also developed Swiss Centre for Innovation in Learning (SCIL) like IceGrain. IceCube is similar to IceGrain but the students produce and comment a single but more complex object. Hence phase 1 and 3 are somehow different from IceGrain.

Phase 1

The students must produce **a single contribution** defined by the teacher. This contribution can include quiz answers, free text and pictures.

Phase 2

Each student has to give feedback and grade other students' contributions.

Phase 3

Groups of students sit together and build together a **group feedback.**

Phase 4

Individual students revise their initial contributions based on the feedback received.

Phase 5

Discussing all contributions and feedback in order to induce new general principles or to revise the initial principles based on this experience.

Choosing an Ice Cube

IceCube is similar to IceGrain but should be applied when contributions are complex objects such as JAVA code, MATLAB scripts, a long text (e.g. a scientific paper),… Not only the time for producing these contributions will be longer, but also the time to read them and to provide feedback as well as the time for revising the object. Therefore, it is expected that each individual will only produce **one** contribution.

Scope

This applies to courses which are somewhat project-oriented, i.e. in which the production by students is an important part, and which will probably be graded at the end of the process.

In the script, this contribution is made by individuals but we might imagine situations where the teacher directly enters pairs of students instead of individuals when creating the class. For instance, "Lena & Lone" would be a single user for the machine but two actual students.

Creating the script content:

Editing contents

The preparation of the script is very similar to IceGrain, except that students will upload a single contribution. It is recommended that each student produces two feedbacks because each contribution needs to get at least two feedbacks in order to enable the collaborative feedback revision phase.

Defining the session

The session parameters are the same as for the IceGrain. Please refer to pages 34-43

Running Phase 1:

Individual contribution

This phase is the same as for IceGrain.

Running Phase 2:

Feedback

This phase is the same as for IceGrain.

Running Phase 3:

Collaborative feedback revision

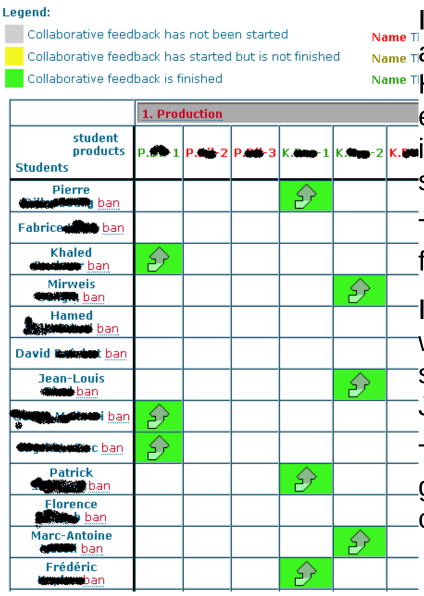
The collaborative feedback revision is a key phase of this script. Students are expected to learn by confronting their opinion with the opinions of other students regarding the same contribution. To do so, students simply sit together in front of the system which displays their individual feedback. It is presented in the same way as in the individual feedback revision phase of IceGrain.

In our experience, we asked students to make short appointments (15-30 min) with each other for this phase. However, with large classes or when students don't know much each other, it might be more effective to schedule all these interactions during one class-contact hour hoping that all student who have to meet will find the possibility to do so.

The assignment of group feedback proceeds as for individual feedback.

In the example below Khaled's first contribution (column K1) will be reviewed by Pierre, Patrick and Frédéric while his second contribution (column K2) will be reviewed by Mirweis, Jean-Louis and Marc-Antoine.

The assignments can be modified by the teacher but cells in green indicate that the joint feedback has already been completed. The yellow colour indicates a feedback in progress.



Legend:

⬜ Collaborative feedback has not been started

🟨 Collaborative feedback has started but is not finished

🟩 Collaborative feedback is finished

Name Tl

Name Tl

Name Tl

	1. Production					
student products Students	P.[redacted]1	P.[redacted]2	P.[redacted]3	K.[redacted]1	K.[redacted]2	K.[redacted]
Pierre [redacted] ban				🟩		
Fabrice [redacted] ban						
Khaled [redacted] ban	🟩					
Mirweis [redacted] ban					🟩	
Hamed [redacted] ban						
David [redacted] ban						
Jean-Louis [redacted] ban					🟩	
[redacted] ban	🟩					
[redacted] ban	🟩					
Patrick [redacted] ban				🟩		
Florence [redacted] ban						
Marc-Antoine [redacted] ban					🟩	
Frédéric [redacted] ban				🟩		

Running Phase 4:

Contribution revision

This phase is the same as for IceGrain.

Running Phase 5:

Debriefing

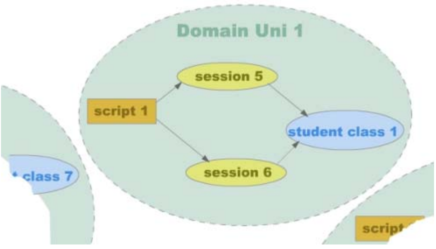
This phase is the same as for IceGrain.

7. Managing scripts

This section explains how to apply different scripts to the same class, how to manage different instances of a script, how to borrow/share scripts from/with colleagues, etc.

The domains

Scripts, sessions, and student classes are grouped by *domains*. As long as a teacher prepares his or her own scripts (e.g. 2 instances of ArgueGraph and 1 instance of IceCube) for his own students (e.g. 3 different classes every year), all information will be gathered within his domain.

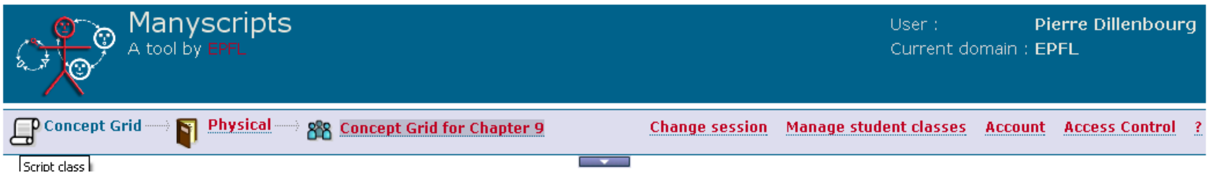


A teacher can belong to more than one domain, but once he or she is connected to a domain, (s)he can only see resources of which they are part.

When the teacher logs in, he or she automatically enters to the last domain in which they were working. To connect to another domain he or she belongs to, she or he needs to use the "change session" menu in the pathbar (just below the banner).

If a teacher belongs to several domains, his or her rights to edit/modify/delete instances or sessions might be different for the domains: one may only delete things that one has created (See the section below on managing rights). Another teacher may not enter your domain unless explicitly invited by the creator of the domain (See section 'sharing scripts').

The ManyScripts banner informs you about the use of the currently selected domain (top right).



Clicking on the ▼ icon just below the middle of the banner develops a complete view of the domain (see next page).

script

script content

session

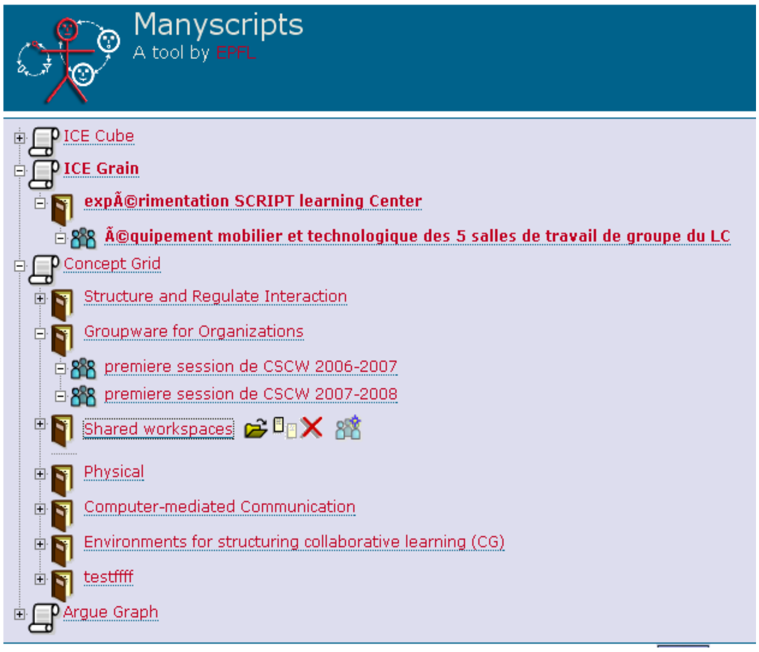
When a button selected, four operations are available:

open and edit the script content

duplicate the script content to make changes without changing the existing version

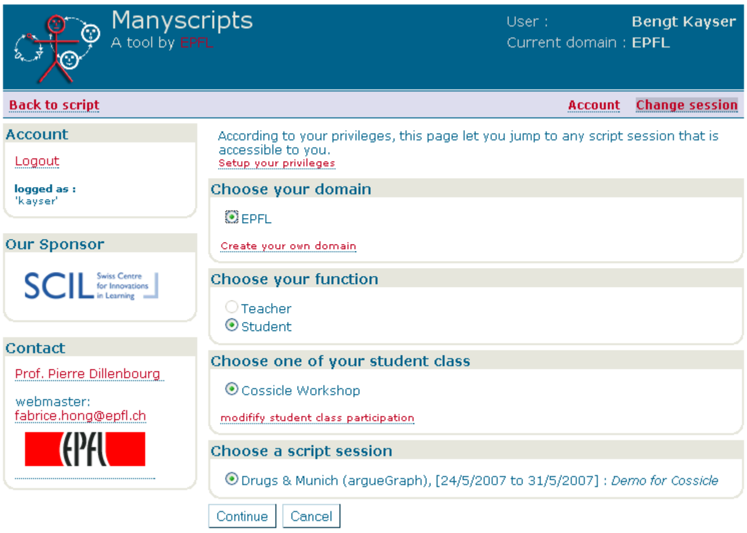
delete the script content

create a session



Manyscripts
A tool by EPFL

- ICE Cube
- ICE Grain
  - expÃ©rimentation SCRIPT learning Center
    - Ã©quipement mobilier et technologique des 5 salles de travail de groupe du LC
- Concept Grid
  - Structure and Regulate Interaction
  - Groupware for Organizations
    - premiere session de CSCW 2006-2007
    - premiere session de CSCW 2007-2008
  - Shared workspaces
  - Physical
  - Computer-mediated Communication
  - Environments for structuring collaborative learning (CG)
  - testffff
- Argue Graph



Manyscripts
A tool by EPFL

User : Bengt Kayser
Current domain : EPFL

Back to script Account Change session

Account
Logout
logged as : 'kayser'

Our Sponsor
SCIL Swiss Centre for Innovations in Learning

Contact
Prof. Pierre Dillenbourg
webmaster: fabrice.hong@epfl.ch
EPFL

According to your privileges, this page let you jump to any script session that is accessible to you.
Setup your privileges

Choose your domain
● EPFL
Create your own domain

Choose your function
○ Teacher
● Student

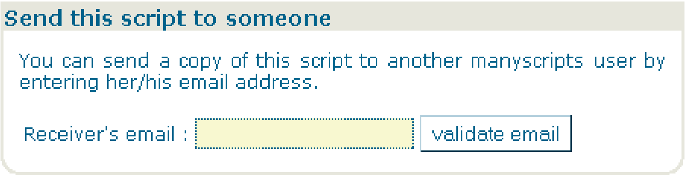
Choose one of your student class
● Cosside Workshop
modifify student class participation

Choose a script session
● Drugs & Munich (argueGraph), [24/5/2007 to 31/5/2007] : *Demo for Cosside*

Continue Cancel

Sharing scripts with colleagues

If a colleague is already a registered user of ManyScripts, you may share with him a copy of the script contents. To do so, you have to enter the email address by which he or she registered in ManyScripts and the instance will be duplicated in one of his domains. He or she has then full rights to edit or delete this script, without affecting your own script.



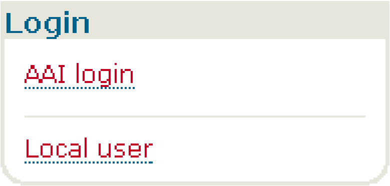
Send this script to someone

You can send a copy of this script to another manyscripts user by entering her/his email address.

Receiver's email : validate email

Login and user account

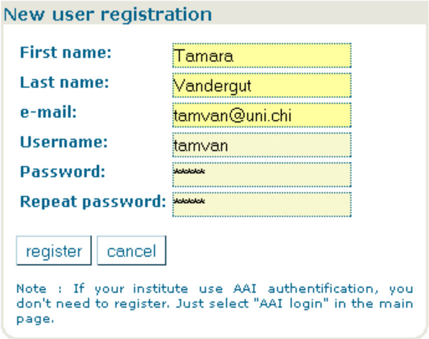
Two forms of registration are enabled. If the user is member of a Swiss University, he or she may use his or her



Login

university username and password by using the "AAI Login" option. This avoids having to remember a specific account and is based on a Swiss 'single login' policy in academic institutions. Otherwise, the user will define his or her own user account (snapshot below, left).

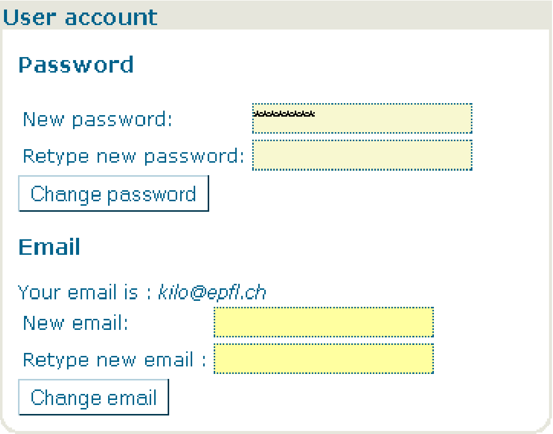
Information on users can be modified at any time by using the "account" item in the top banner of ManyScripts (See snapshot below, right)



New user registration

First name:	Tamara
Last name:	Vandergut
e-mail:	tamvan@uni.chi
Username:	tamvan
Password:	*****
Repeat password:	*****

Note : If your institute use AAI authentification, you don't need to register. Just select "AAI login" in the main page.



User account

Password

New password: ********

Retype new password:

Email

Your email is : *kilo@epfl.ch*

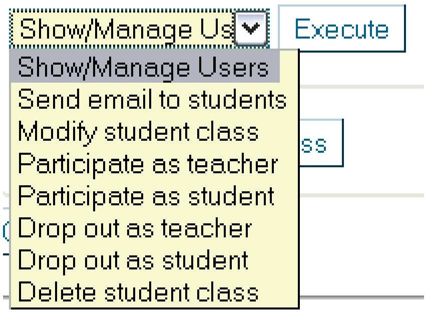
New email:

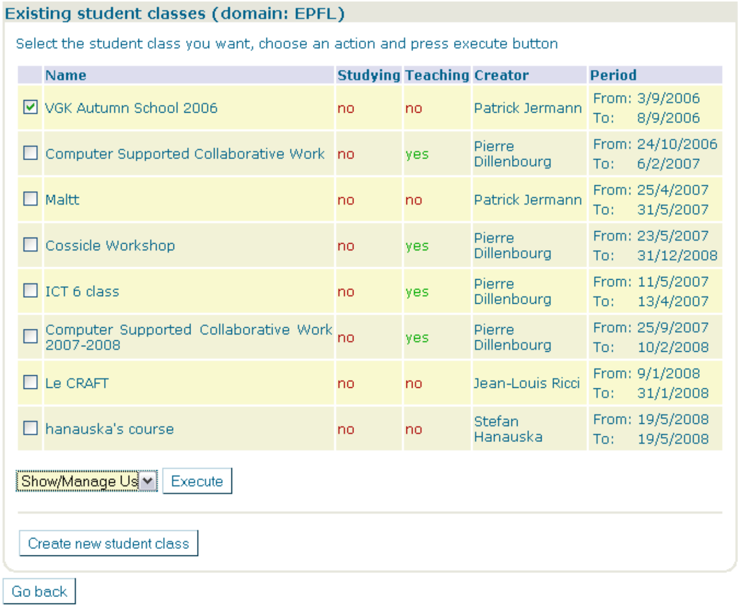
Retype new email :

Managing students

The teacher is not expected to add students by hand into some database, but let the students simply register themselves and choose the course and the script session to which they have been asked to join.

ManyScripts is not a full learning management system but includes a few functionalities for managing students classes: to create/delete a class, to add/remove a teacher to/from a class, to remove a student from a class. The "teacher" role is also used by assistants who help the teacher to run the script.





Existing student classes (domain: EPFL)

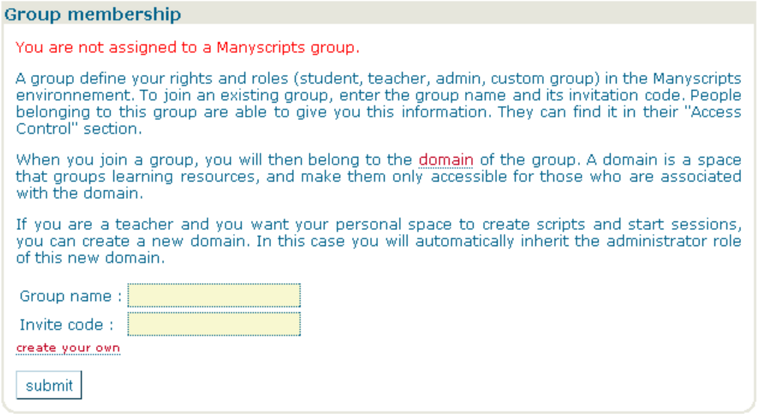
Select the student class you want, choose an action and press execute button

	Name	Studying	Teaching	Creator	Period
☑	VGK Autumn School 2006	no	no	Patrick Jermann	From: 3/9/2006 To: 8/9/2006
□	Computer Supported Collaborative Work	no	yes	Pierre Dillenbourg	From: 24/10/2006 To: 6/2/2007
□	Maltt	no	no	Patrick Jermann	From: 25/4/2007 To: 31/5/2007
□	Cossicle Workshop	no	yes	Pierre Dillenbourg	From: 23/5/2007 To: 31/12/2008
□	ICT 6 class	no	yes	Pierre Dillenbourg	From: 11/5/2007 To: 13/4/2007
□	Computer Supported Collaborative Work 2007-2008	no	yes	Pierre Dillenbourg	From: 25/9/2007 To: 10/2/2008
□	Le CRAFT	no	no	Jean-Louis Ricci	From: 9/1/2008 To: 31/1/2008
□	hanauska's course	no	no	Stefan Hanauska	From: 19/5/2008 To: 19/5/2008

A new user will be asked to join some existing group. A student of your course or an assistant will then type the name of the group and an invitation code that were given to them. The name of the group and the system-generated invitation code are displayed when the teacher chooses

"Access Control" in the top banner (see also the "Managing Rights " section below. If you want to create a new domain (new script, new class) select the "Create your own" option at the bottom of the page.

Once logged into ManyScripts, the user is asked to choose his or her role as well as the class to join within the selected domain.



Group membership

You are not assigned to a Manyscripts group.

A group define your rights and roles (student, teacher, admin, custom group) in the Manyscripts environnement. To join an existing group, enter the group name and its invitation code. People belonging to this group are able to give you this information. They can find it in their "Access Control" section.

When you join a group, you will then belong to the domain of the group. A domain is a space that groups learning resources, and make them only accessible for those who are associated with the domain.

If you are a teacher and you want your personal space to create scripts and start sessions, you can create a new domain. In this case you will automatically inherit the administrator role of this new domain.

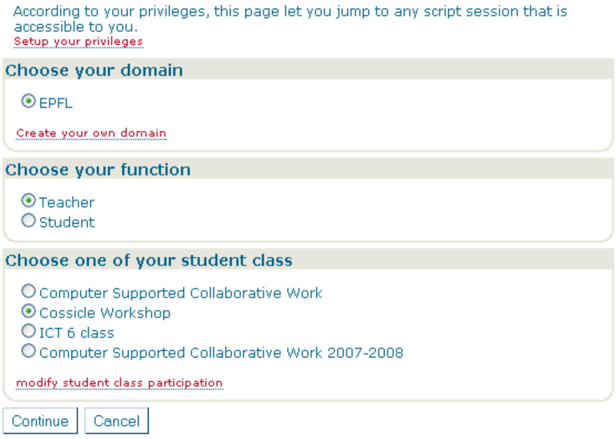
Group name :

Invite code :

create your own

Managing sessions

At any time, the user may change his participation in a script by using the "change session" option in the top banner. This enables a student to participate in 2 scripts from different courses at the same time and to switch between them.



According to your privileges, this page let you jump to any script session that is accessible to you.
Setup your privileges

Choose your domain

● EPFL

Create your own domain

Choose your function

● Teacher
○ Student

Choose one of your student class

○ Computer Supported Collaborative Work
● Cosside Workshop
○ ICT 6 class
○ Computer Supported Collaborative Work 2007-2008

modify student class participation

Managing users

Participation can be modified by choosing the "Access control" item in the top banner. ManyScripts users may have 3 different roles, i.e. levels of rights for editing scripts contents, sessions or classes:

Domain admin

Users with this role can modify or delete any script content, sessions and student class in the domain.

Prof

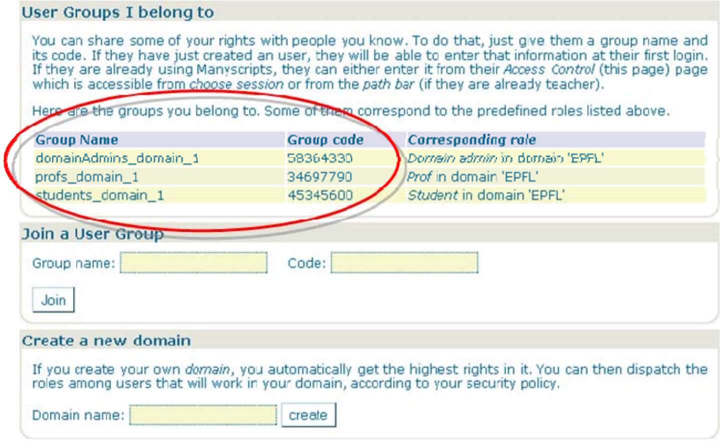
Users with this role can create new script content, sessions and student classes in the domain. They can use any scripts to create their own sessions in the domain. They can modify or delete only the object that they created in the domain.

Student

Users with this role can only play a script. Typically, the teacher sends the group name and code to the students so that they can have this role, and thus run the script.

A user may join an existing user group in an existing domain, create a new domain or invite people to join his or her own users' group in a domain.

The information circle in red is the information to be communicated to students who have to join your domain and script session.



User Groups I belong to

You can share some of your rights with people you know. To do that, just give them a group name and its code. If they have just created an user, they will be able to enter that information at their first login. If they are already using Manyscripts, they can either enter it from their *Access Control* (this page) page which is accessible from *choose session* or from the *path bar* (if they are already teacher).

Here are the groups you belong to. Some of them correspond to the predefined roles listed above.

Group Name	Group code	Corresponding role
domainAdmins_domain_1	58364330	*Domain admin* in domain 'EPFL'
profs_domain_1	34697790	*Prof* in domain 'EPFL'
students_domain_1	45345600	*Student* in domain 'EPFL'

Join a User Group

Group name:

Code:

Create a new domain

If you create your own *domain*, you automatically get the highest rights in it. You can then dispatch the roles among users that will work in your domain, according to your security policy.

Domain name:

Inquiries can be sent to pierre.dillenbourg@epfl.ch and taiga.brahm@unisg.ch

Literature

In the following, you will find literature that might be of interest for you when using scripts or the underlying ideas provided in this pedagogical handbook.

Arvaja, M., Salovaara, H., Häkkinen, P. et al. (2007). Combining individual and group-level perspectives for studying collaborative knowledge construction in context. *Learning and Instruction, 17*, 448-459.

Carell, A. (2006). *Selbststeuerung und Partizipation beim computergestützten kollaborativen Lernen: Eine Analyse des Zusammenhangs von Selbststeuerung und Partizipation im Kontext hochschulischer Lernprozesse*. Münster: Waxmann.

Cress, U. & Kimmerle, J. (2008). A systemic and cognitive view on collaborative knowledge building with wikis. *Computer-Supported Collaborative Learning, 3*, 105-122.

Dillenbourg, P. (1999). What do you mean by 'collaborative learning'? In P. Dillenbourg (Hrsg.), *Collaborative-learning: Cognitive and Computational Approaches* (S. 1-19). Oxford: Elsevier.

Dillenbourg, P. (2002). Over-scripting CSCL: The risks of blending collaborative learning with instructional design. . In P. A. Kirschner (Hrsg.), *Three worlds of CSCL. Can we support CSCL?* (S. 61-91). Heerlen: Open Universiteit Nederland.

Dillenbourg, P. & Fischer, F. (2007). Basics of Computer-Supported Collaborative Learning. In D. Euler et al. (Hrsg.), *Kooperatives Lernen in der beruflichen Bildung* (21, S. 111-130). Stuttgart: Franz Steiner Verlag.

Dillenbourg, P. & Jermann, P. (2006). Designing Integrative Scripts. In F. Fischer et al. (Hrsg.), *Scripting Computer-Supported Collaborative Learning: Cognitive, Computational, and Educational Perspectives* (S. 275-302). New York et al.: Springer.

Dillenbourg, P. & Tchounikine, P. (2007). Flexibility in macro-scripts for computer-supported collaborative learning. *Journal of Computer Assisted Learning, 23* (1), 1-13.

Hogarth, A. (2008). Introducing a collaborative technology strategy for higher education students: Recommendations for the way foward. *Education and Information Technologies, 13* (3), 259-273.

Rummel, N. & Spada, H. (2005). Learning to Collaborate: An Instructional Approach to Promoting Collaborative Problem Solving in Computer-Mediated Settings. *The Journal of the Learning Sciences, 14* (2), 201-241.

Rummel, N., Spada, H. & Hauser, S. (2009). Learning to collaborate while being scripted or by observing a model. *Computer-Supported Collaborative Learning, 4* (1), 69-92.

Seufert, S. & Brahm, T. (Hrsg.) (2007), *"Ne(x)t Generation Learning": Wikis, Blogs, Mediacasts & Co. - Social Software und Personal Broadcasting auf der Spur*. St. Gallen: scil, Universität St. Gallen.

Swiss Centre for Innovations in Learning (scil)

University of St. Gallen

Dufourstrasse 40a
CH-9000 St. Gallen

Telefon: +41 (0)71 224 31 55
Telefax: +41 (0)71 224 26 19

E-Mail: scil-info@unisg.ch
Internet: http://www.scil.ch

About scil
The Swiss Centre for Innovations in Learning (scil) promotes the competent and meaningful use of new technologies in university and corporate education. scil provides consulting, coaching, research, and moderation to accelerate progress towards enhanced quality in education. The centre was founded in March 2003. It is supported by the GEBERT RÜF FOUNDATION.