

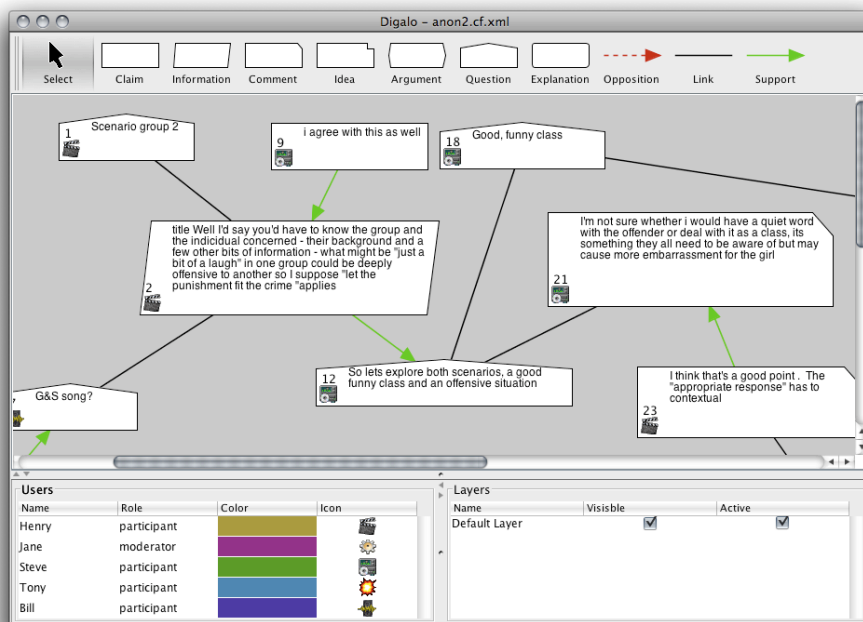
# Overview of the Digalo discussion environment within the Argonaut System

Matthias Krauß, Fraunhofer IAIS.VE

Version 1.1, 13-Jan-2008

## Introduction

Digalo is a tool for supporting argumentative discussions. It uses graphic means for this purpose, employing a graph structure. In a discussion, graph nodes represent discussion contributions and edges express relations between contributions. Participants discuss by adding, editing, moving and deleting nodes (“*Shapes*”) and edges (“*Links*”) in a graph shared between all participants (“*Map*”). Digalo’s fields of application are numerous and varied. One main field of usage is in learning or education settings.



The initial version of Digalo was developed during the *Dunes*<sup>1</sup> project. During the <sup>2</sup>project, Digalo was substantially consolidated and updated to current IT technology standards and requirements. At the same time, obsolete parts and dependencies were removed and functionality needed for compatibility with the Argonaut infrastructure was added. Error-prone and inefficient parts were rewritten. Due to the extent of changes in its code base, Digalo evolved to Digalo2.

Still, Digalo2 was designed to embrace users of Digalo1 – the user interface was simplified but its concepts either remained in its original form or were replaced in a similar, but more powerful way.

The remainder of this text primarily describes the newer version, hinting to differences from the original version. Wherever the versions need to be distinguished, the terms *Digalo2* and *Digalo1* are used, respectively.

### **Single- and multi-user mode**

As a standalone application, Digalo works in two modes: Single-user (“*Local*”) and multi-user (“*synchronous*”) mode. In local mode, a single user may operate on a map stored in a local file. In synchronous mode, multiple users are working on a shared map, stored and coordinated through a shared Digalo server. Digalo1 used multiple communication techniques to overcome different technical networking obstacles such as firewalls. Digalo2’s uses only one networking technique that does not expose any of these problems.

### **User roles**

Digalo distinguishes between two types of users: participants and facilitators. Participants (usually students in a school environment) have limited access privileges and no access to the map’s administration. Facilitators (usually teachers and/or dedicated students in a school environment) have full editing access and may change the map’s settings, such as its ontology, layers, privileges, users and background image and/or grid (see below).

### **Joining a map**

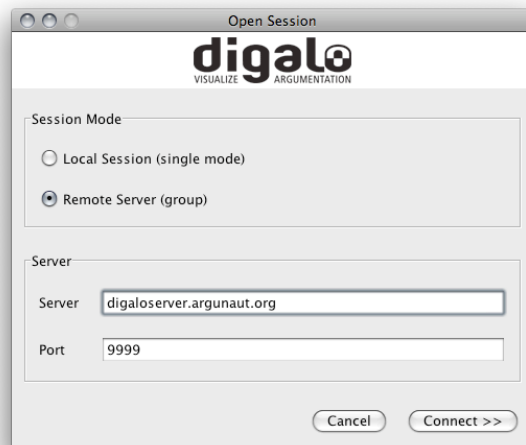
By default, Digalo guides the user to its map through a three-step process on startup:

---

<sup>1</sup> *DUNES – Dialogic and argUmentative Negotiation Educational Software*, March 1, 2002 to August 31, 2004. Project Reference: IST-2001-34153

<sup>2</sup> *ARGUNAUT – An Intelligent Guide to Support Productive Online Dialogue*, December 1, 2005 to August 31, 2008. Project Reference: FP6-IST 027728

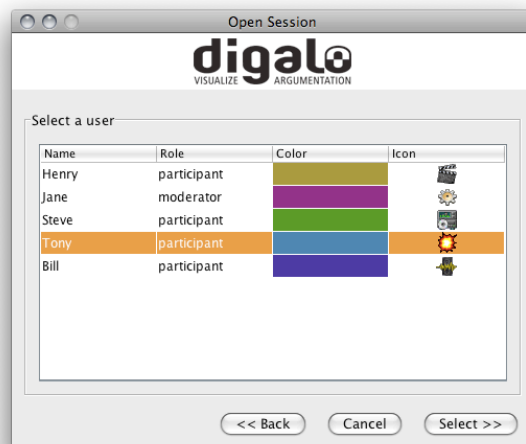
1. Select the server:



2. Select the map:



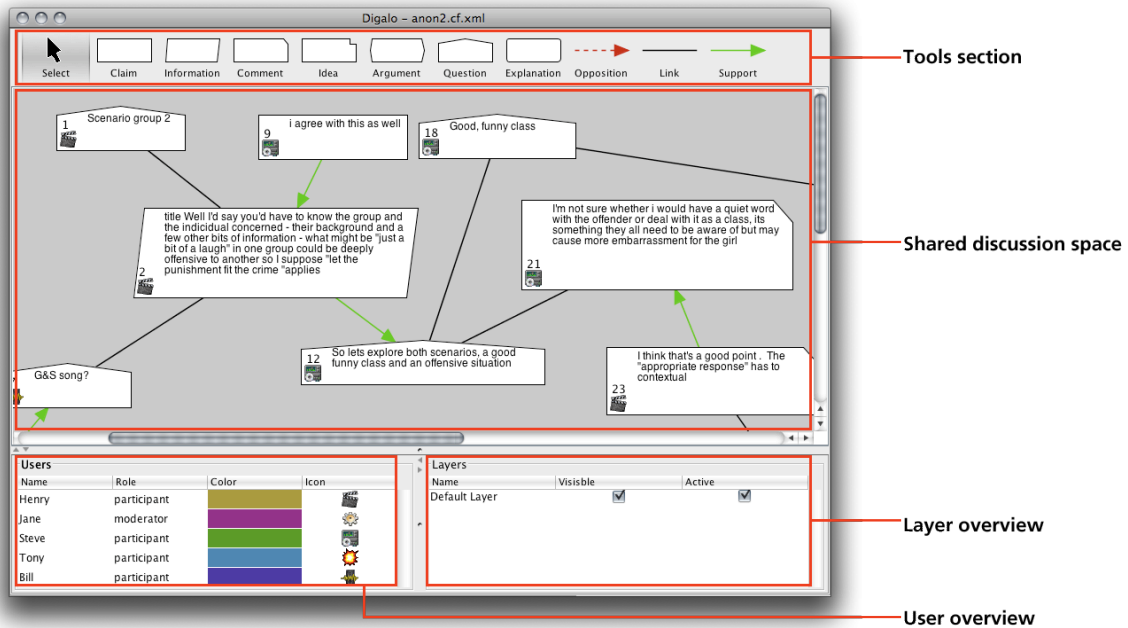
3. Select the user:



However, Digalo may be started with preconfigured server, map and/or user. If all needed parameters are given, Digalo starts directly into the running discussion – e.g. when launched via web start or programatically in an environment such as Argonaut.

### Participating in a discussion

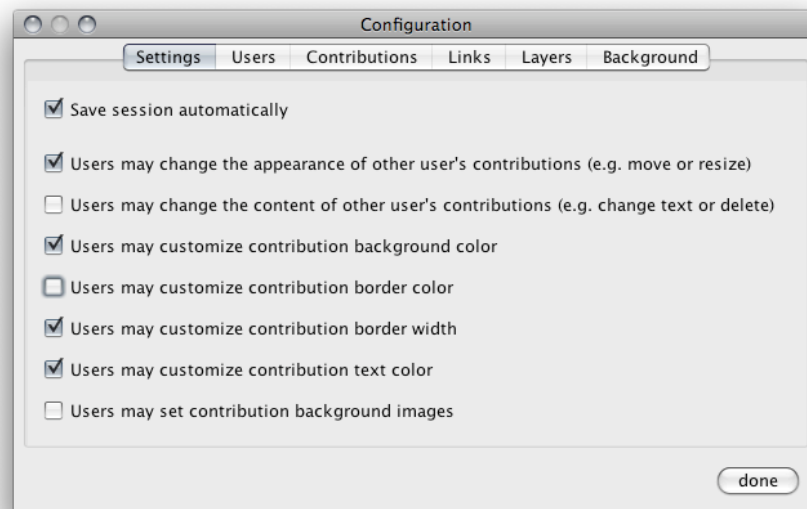
Digalo2's main screen has been significantly simplified. In its purest form, it just shows the shared discussion space and a toolbar. Additionally, users and layers can be shown at the bottom:



Participants add their contributions by selecting an available shape and placing it into the map. With the selection tool, users may select, move, resize and inspect contributions. Hovering the mouse cursor over a contribution exposes a *tooltip* window with contribution details not visible by default. Freshly added and double-clicked contributions open a contribution editor dialog, if allowed:



Users may edit contribution contents and appearance. Facilitators may configure access rules, e.g. whether users are allowed to change other participant's contribution contents. Digalo2 offers more fine-grained control over these access rights:



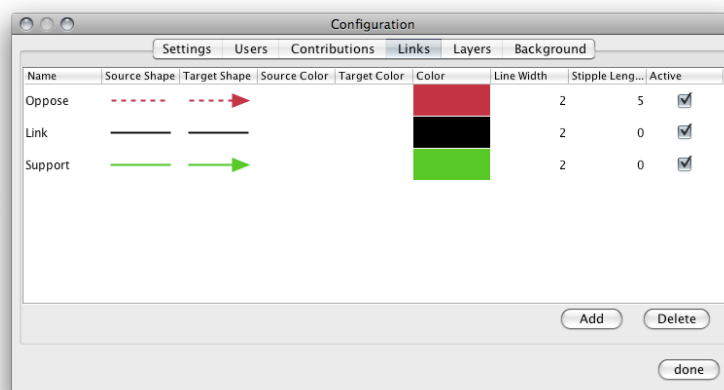
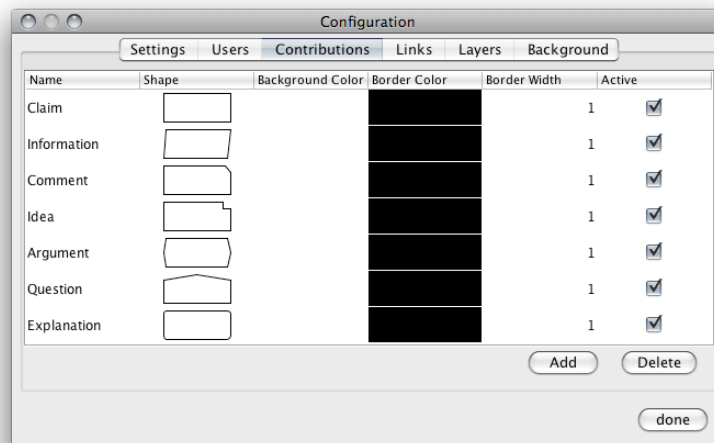
### **Title and Note**

Contributions have two text fields for their content: The *Title* is supposed to hold a short summary or title of the contribution. Its content is always visible in the map view. In contrast, the *Note* field may hold a more detailed description. Its content is not visible by default – in order to see it, a user must either open the contribution inspector or hover the mouse over the contribution to see its tooltip. This separation of content fields allows expressive and detailed contributions to be made without cluttering the map display or requiring overly large contribution shapes. Digalo1 did not reveal the full contribution's content in its tooltip – users had to right-click the contribution in order to see it. Since this behaviour was often not figured out by users, the tooltip shows a formatted and easy-to-read description in the standard tooltip.

### **Ontology**

The ontology – represented here by shapes and arrows – is aimed at facilitating the acquiring of argumentative skills.

The facilitator can choose which shapes would appear in each map s/he creates using the following default list:



Shapes' names can be changed, as well as the language used in the ontology. Only the discussion facilitator is allowed to make such changes.

Whereas Digalo1 only allowed up to seven contribution types (one per predefined shape) and could not cope with ontology changes during a discussion, Digalo2 allows an unlimited number of contribution and link types (predefined shapes plus color and line variations) and allows the shapes to be changed during a discussion. This way, facilitators may employ more fine-grained variations of contribution types or introduce different discussion phases with different contribution types available, e.g. a peer review phase after the actual discussion.

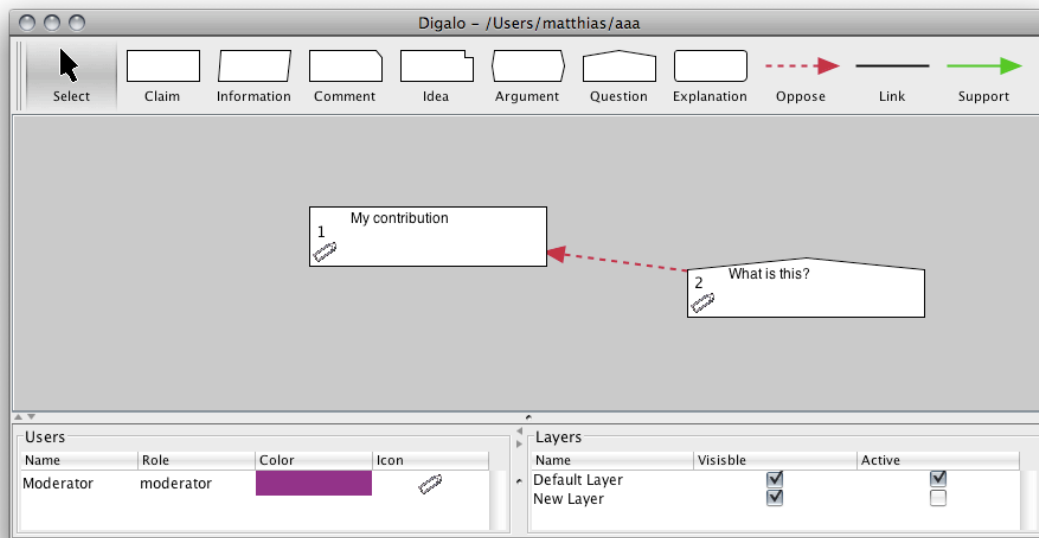
### Floor Control

Digalo1 employed an access mode called *Floor Control*, where only one participant (the floor owner) was allowed to contribute to a discussion at a time. All other participants had to wait for their time slot. This mode was mainly a historical reminder of early Digalo development phases during the Dunes project. As this mode was rarely used and mainly caused confusion among participants as well as facilitators, it was removed. In Digalo2, the general mode of operation is free access: All participants are allowed to contribute simul-

taneously. More fine-grained control of operation can be accomplished through of Argunaut's moderation facilities.

## Layers

Digalo allows the facilitator to define multiple layers. Each contribution is assigned to one specific layer. Layer visibility may be turned on and off individually to simplify large discussions or to distinguish discussion phases. Each user may configure layer visibility individually. A bar at the bottom of the main screen may be expanded and collapsed for this purpose:



## Map canvas, grid and background image

In Digalo, the map background is a virtual empty area of theoretically unlimited size, expanding and shrinking as needed. In Digalo1, maps could only expand to the bottom right, whereas Digalo2 allows a map to grow into all directions. If the map size exceeds the available screen size, Digalo2 allows to choose either to dynamically scale down the graph display to fit the whole map or to show sub-area of fixed size on the screen.

Instead of a plain color, facilitators may set a regular grid or an arbitrary image as map background, for example to illustrate a discussion topic or to provide a spatial structure. In Digalo1, any participant could set the background image, resulting a) in an inconsistency because users could set the image but only facilitators could set a grid and b) sometimes causing users to overwrite each other's background images over and over again. Digalo2 permits the specification of the map background only to facilitators. Instead, users may set image backgrounds to individual contribution shapes (if map settings permit this operation). In Digalo1, no contribution images could be specified.

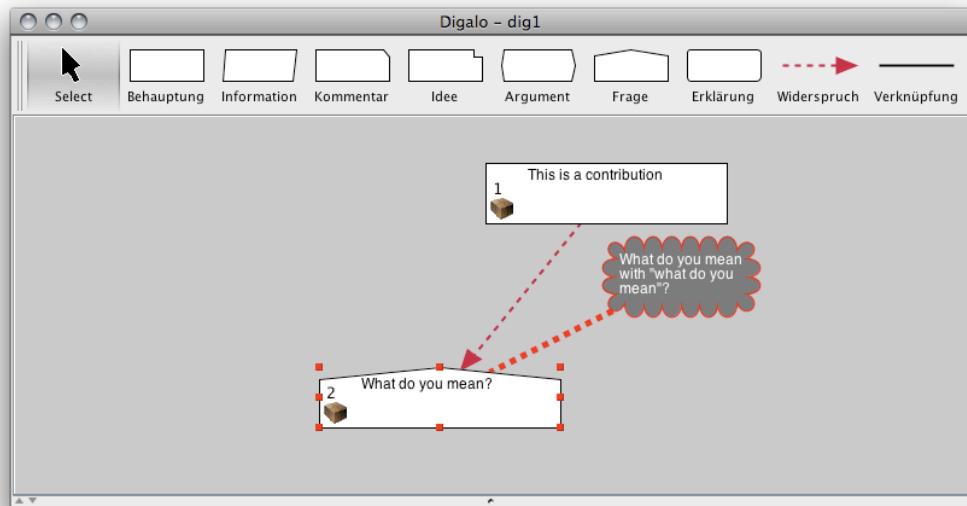
## Other features

Digalo has a set of other features, such as built-in replay, overview, export functions etc. Most of these features already existed in Digalo1. However, as these features are not strictly relevant for understanding the core concepts of the approach, this document does not describe them in detail.

## Digalo in the Argunaut environment

Digalo2 was extended in five ways in order to be integrated into the Argunaut infrastructure:

1. Digalo maps may be directly started from the Argunaut client. This way, users may launch a Digalo session by double-clicking on a map in an Argunaut portfolio.
2. Digalo may show its discussion windows as subwindows of another application instead of standalone windows. This way, the Argunaut client may present Digalo as a plugin, not as a separate standalone application.
3. Digalo2's session and user management was extended to be externally configurable and controllable. The Argunaut system keeps its own list of portfolios, sessions and users. With this extension, Digalo's own session and user management may be updated to reflect the Argunaut portfolio management. In combination with the aforementioned extension, Digalo2's user and session management becomes completely transparent.
4. Digalo2 may forward information about discussion progress to external entities in real time as *Common Format* actions. This way, Argunaut's *Moderator's Interface* may reflect the current status of all active discussions in real time. This extension is eased by the fact that Digalo2 uses the XML-based *Common Format* as its native storage format. Digalo1 used a proprietary, binary, non human-readable and Java version-dependent format to store its maps.
5. Digalo2 may accept and react to *Remote Control* messages from external sources in a session. For example, a contribution may be highlighted, a moving pointer may be shown, contributions may be annotated or a text and/or image message may be displayed. This extension allows the moderator to issue moderation interventions from the *Moderator's Interface's Control Panel*. The following screen shows an annotated contribution:



These extensions allow Digalo2 to be plugged seamlessly into the Argunaut infrastructure. Its nature of being a separate tool seems to disappear: Digalo becomes an Argunaut plugin. Nevertheless, Digalo2 remains to be usable as a standalone application.