The ALMAgame

ALMAgame is a Logging, Monitoring and Alerting game

M. Finelli
BioDec

BIODEC
Evolving ICT Infrastructures

Index

BIODEC
Evolving ICT Infrastructures

Index

BIODEC
Evolving ICT Infrastructures

A game to play with distributed systems

ALMA game is a very simple game where:

- you are an agent (identified by a *name*);

- you spend your time sending and receiving *messages — i.e.* coloured post-it like notes;

- you have one or more *cards* with the criteria used to send and receive messages;

- you have a *bookkeeping log* in the form of a piece of paper, it is divided in two columns and it has numbered lines, one for each turn;

- THAT'S ALL FOLKS !

A game to play with distributed systems

ALMA game is a very simple game where:

- you are an agent (identified by a *name*);

- you spend your time sending and receiving *messages* — *i.e.* coloured post-it like notes;

- you have one or more *cards* with the criteria used to send and receive messages;

- you have a *bookkeeping log* in the form of a piece of paper, it is divided in two columns and it has numbered lines, one for each turn;

- THAT'S ALL FOLKS !

BIODEC
Evolving ICT Infrastructures

A game to play with distributed systems

ALMA game is a very simple game where:

- you are an agent (identified by a *name*);
- you spend your time sending and receiving *messages* — *i.e.* coloured post-it like notes;
- you have one or more *cards* with the criteria used to send and receive messages;
- you have a *bookkeeping log* in the form of a piece of paper, it is divided in two columns and it has numbered lines, one for each turn;
- THAT'S ALL FOLKS !

BIODEC
Evolving ICT Infrastructures

A game to play with distributed systems

ALMA game is a very simple game where:

- you are an agent (identified by a *name*);
- you spend your time sending and receiving *messages* — *i.e.* coloured post-it like notes;
- you have one or more *cards* with the criteria used to send and receive messages;
- you have a *bookkeeping log* in the form of a piece of paper, it is divided in two columns and it has numbered lines, one for each turn;
- THAT'S ALL FOLKS !

A game to play with distributed systems

ALMA game is a very simple game where:

- you are an agent (identified by a *name*);

- you spend your time sending and receiving *messages* — *i.e.* coloured post-it like notes;

- you have one or more *cards* with the criteria used to send and receive messages;

- you have a *bookkeeping log* in the form of a piece of paper, it is divided in two columns and it has numbered lines, one for each turn;

- THAT'S ALL FOLKS !

BIODEC
Evolving ICT Infrastructures

A game to play with distributed systems

ALMA game is a very simple game where:

- you are an agent (identified by a *name*);
- you spend your time sending and receiving *messages* — *i.e.* coloured post-it like notes;
- you have one or more *cards* with the criteria used to send and receive messages;
- you have a *bookkeeping log* in the form of a piece of paper, it is divided in two columns and it has numbered lines, one for each turn;
- THAT'S ALL FOLKS !

BIODEC
Evolving ICT Infrastructures

Index

Turns

The game is played in *turns*. Each turn read your cards and play accordingly. Each turn has the following phases:

1. sending,
2. bookkeeping,
3. referee's intervention,
4. receiving,
5. bookkeeping.

Turns

Send phase First, *send* a message to somebody putting it in front of you, directed to the destination. Your cards may dictate to whom you have to send messages: for example a card may state *"Send three messages, but one of them always to agent A"*.

Bookkeping Annotate on your log, in the SEND column, the name of the destination: *one message = one name*.

Turns

Referee phase Await for the *referee* to act: he can take any
message on the board and throw it away.

Receive phase Next, *receive* any (even zero!) of the messages
sent to you. Your cards may restrict your freedom to
take the messages: for example a card may state *"You
can receive at most one message"*.

Bookkeping Annotate in your log, in the RECEIVE column, the
name of the source: *one message = one name*.

The log format

Let's say player Z sends two messages to A the first turn, and receives nothing; next he sends a message to A and B each, and receives a message from both of them; on the third turn he sends nothing and receives just a message from A.

Zed's turn	SEND	RECEIVE
1.	A,A	-
2.	A,B	A,B
3.	-	A

The log format

Let's say player Z sends two messages to A the first turn, and
receives nothing; next he sends a message to A and B each, and
receives a message from both of them; on the third turn he sends
nothing and receives just a message from A.

Zed's turn	SEND	RECEIVE
1.	A,A	-
2.	A,B	A,B
3.	-	A

The log format

Let's say player Z sends two messages to A the first turn, and receives nothing; next he sends a message to A and B each, and receives a message from both of them; on the third turn he sends nothing and receives just a message from A.

Zed's turn	SEND	RECEIVE
1.	A,A	-
2.	A,B	A,B
3.	-	A

The log format

Let's say player Z sends two messages to A the first turn, and receives nothing; next he sends a message to A and B each, and receives a message from both of them; on the third turn he sends nothing and receives just a message from A.

Zed's turn	SEND	RECEIVE
1.	A,A	-
2.	A,B	A,B
3.	-	A

The log format

Let's say player Z sends two messages to A the first turn, and receives nothing; next he sends a message to A and B each, and receives a message from both of them; on the third turn he sends nothing and receives just a message from A.

Zed's turn	SEND	RECEIVE
1.	A,A	-
2.	A,B	A,B
3.	-	A

The log format

Let's say player Z sends two messages to A the first turn, and receives nothing; next he sends a message to A and B each, and receives a message from both of them; on the third turn he sends nothing and receives just a message from A.

Zed's turn	SEND	RECEIVE
1.	A,A	-
2.	A,B	A,B
3.	-	A

The log format

Let's say player Z sends two messages to A the first turn, and receives nothing; next he sends a message to A and B each, and receives a message from both of them; on the third turn he sends nothing and receives just a message from A.

Zed's turn	SEND	RECEIVE
1.	A,A	-
2.	A,B	A,B
3.	-	A

Let's play !

Index

Coloured messages

As you may have noticed, everybody has its own set of post-it like notes. Each set has a *distinguished colour*: let's use it to add a couple of rules to the sending phase.

The coloured messages rules

Send your own You can send your own post-it likes notes (*i.e.* those with your own colour) to anybody.

Reply to the owner Once you have received a messages from somebody, you can do anything you want with it — even throw it away — but if you send it back, *you can only send it back to its owner*.

BIODEC
Evolving ICT Infrastructures

The bookkeping

If you get back one of your own messages (*i.e.* a post-it like note of your own colour) and you decide to receive it, you are required to mark that fact in the log: just draw a circle around the letter to distinguish them from ordinary messages.

Zed's turn	SEND	RECEIVE
1.	B,B	-
2.	A,B	A,Ⓑ
3.	-	A

On the second turn Z gets back from B one of its own messages.

The bookkeping

If you get back one of your own messages (*i.e.* a post-it like note of
your own colour) and you decide to receive it, you are required to
mark that fact in the log: just draw a circle around the letter to
distinguish them from ordinary messages.

Zed's turn	SEND	RECEIVE
1.	B,B	-
2.	A,B	A,Ⓑ
3.	-	A

On the second turn Z gets back from B one of its own messages.

The idea behind the coloured messages

Since a Ⓑ can only appear in the log of Z if B replies to a message that he or she has already received from Z, Ⓑ can be used to model an *acknowledge* to some previous message.

More complicated communications

In this way we can realize simple communication patterns, for example a sort of **"ping protocol"**.

We have a way on learning that an host is alive by sending to it a message (modelling an ECHO REQUEST, in ping-tongue) and awaiting for a reply of the *same colour* (modelling an ECHO REPLY).

Let's play !

Index

Introduce the roles

Now, each player gets a card. The card gives a *role*:

- the web server;
- the database server;
- the mail server;
- the backup server;
- the master server.

According to the role there are *rules* for sending and receiving messages, and a condition that triggers an *alert*.

BIODEC
Evolving ICT Infrastructures

Michele Finelli   DevOps

Introduce the alert

Some of the rules say that when something happens you enter an *alert state*. What does it mean exactly ?

Nothing ! Just take something silly, like a funny hat, a toddler's toy or else, and put in front of you, screaming:

ALARM ! I AM IN ALARM !

Just once. Really: the turn after you have entered an alert state, *silence it.*

Introduce the alert

Some of the rules say that when something happens you enter an *alert state*. What does it mean exactly ?
Nothing ! Just take something silly, like a funny hat, a toddler's toy or else, and put in front of you, screaming:

ALARM ! I AM IN ALARM !

Just once. Really: the turn after you have entered an alert state, *silence it.*

BIODEC
Evolving ICT Infrastructures

Introduce the alert

Some of the rules say that when something happens you enter an *alert state*. What does it mean exactly ?
Nothing ! Just take something silly, like a funny hat, a toddler's toy or else, and put in front of you, screaming:

ALARM ! I AM IN ALARM !

Just once. Really: the turn after you have entered an alert state, *silence it.*

Introduce the alert

Some of the rules say that when something happens you enter an
alert state. What does it mean exactly ?
Nothing ! Just take something silly, like a funny hat, a toddler's toy or
else, and put in front of you, screaming:

ALARM ! I AM IN ALARM !

Just once. Really: the turn after you have entered an alert state,
silence it.

BIODEC
Evolving ICT Infrastructures

Introduce the alert

Some of the rules say that when something happens you enter an
alert state. What does it mean exactly ?
Nothing ! Just take something silly, like a funny hat, a toddler's toy or
else, and put in front of you, screaming:

<div align="center">

ALARM ! I AM IN ALARM !

</div>

Just once. Really: the turn after you have entered an alert state,
silence it.

The modified bookkeping phase

Bookkeping
1. Annotate in your log, in the RECEIVE column, the name of the source: *one message = one name*.
2. Then, if you were in an alert state, silence it (putting the alarm away).
3. Last, check if any alert condition is satisfied, and eventually put the alarm on again.

Now you have a goal !

Since we have alerts, now *you* have a goal:

try not to raise an alarm.

Beware: during the game you may get *additional cards* that will make your goal impossible to achieve: that's life !

Now you have a goal !

Since we have alerts, now *you* have a goal:

try not to raise an alarm.

Beware: during the game you may get *additional cards* that will
make your goal impossible to achieve: that's life !

Now you have a goal !

Since we have alerts, now *you* have a goal:

try not to raise an alarm.

Beware: during the game you may get *additional cards* that will make your goal impossible to achieve: that's life !

BIODEC
Evolving ICT Infrastructures

Let's play !

Index

A taxonomy of information

		TYPE	
		Events	**States**
ACCESS	**Implicit**	Logging	Alert
	Explicit	Monitoring	