

# k-nase / moomie

- A proposal for developing
- object-oriented messaging middleware
- to support
- e-learning challenges and activities
- - Mike Malloch
- (e-compete partners' meeting, April 22/23 2004)

# Overview

- Background & origins of the project
- Why is it needed?
- What is it?
- How will we develop it?
- Schedule of milestones and steps
- Phase I: open stories - team-tasks + mindnotes

# but first...what do we call it? :O)

- kinase: (biology) A crucial background processor supporting distributed inter/intra-cellular signaling
- k-nase: **K**nowledge **A**rchitecture to **S**upport **e**-learning
- moomie: **m**iddleware for **o**bject-**o**riented **m**essaging in **e**-learning
- The names are provisional :o)

# Background & origins

- KnowNet: team tasks & learning journeys
- NewMediaLab: Platform X
- RayCom: MindNotes, Room-80
- Early IMS standard proposal (Steve Griffin)
  - implemented to crude JAVA-only demo
  - dropped in the 'vendor purges'
- IMS Learn-Design: interoperating learning activities (in bulk, not fine grained)

# Why is it needed?

- e-learning should provide...
  - supported learning challenges
  - not just information
- but good challenges require deep interactions
  - shared, stepped, reflective activity
  - acutely delivered advice & knowledge
- And that is hard!

# v v ny... (4) - common interactivity requirements

- sharing activities & goals in real time
- capturing actions/patterns/achievements
- creating sub-goals / hypotheses etc...
- reflecting, together, on the above
- getting pertinent assistance when needed
- reporting: to self/group/tutor
- talking in context; sharing context in talk

# Why... (3) - Interactivity is hard to achieve

- HTML/Flash etc are designed to present content in solitary sequence
- Though the requirements are common to any prospective e-learning challenge or activity, there is no common support 'middleware' for these, and no-one developing it because...
  - Few developers 'get' e-learning
  - Few teachers 'get' software development

# Why... (4) - The time is ripe & we can do it!

- Since late 90's, major advances in
  - web standards for smart web clients
  - web standards for thin, fast XML data
  - Flash: MX/COM; JAVA: applets/servelets
  - Open source communities & support
- We know-how (and know-who :o) to do this
  - Deep experience, good contacts

# What is it? Two parts:

- MOOMIE - object-messaging service

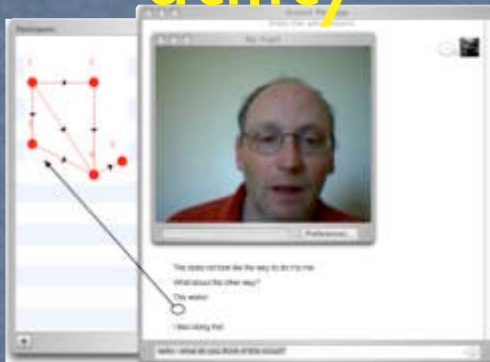


MOOMIE brokers messages

- k-nase - common services and utilities

MOOMIE invokes k-nase services

event-aware utility



remote service



knowledgebase

# What is it... (4) - MOOMIE object

## messaging

- Objects register their events, interests, triggers and handlers; moomie brokers:
- Several kinds of object<>object interaction
  - instance<>instance (same application / several users)
  - instance<>clientside-service (eg event-aware chatboard)
  - instance<>remote-service (eg knowledge snippet repository )

# What is it... (3) - k-nase common services

- Two kinds of service:
  - clientside utilities (with user interfaces)
  - remote knowledgebases, search etc
- Clientside services will provide excellent implementations of common collaboration tools
- Remote services will provide access to acute, fine-grained, embeddable support materials
- Standards implications for both

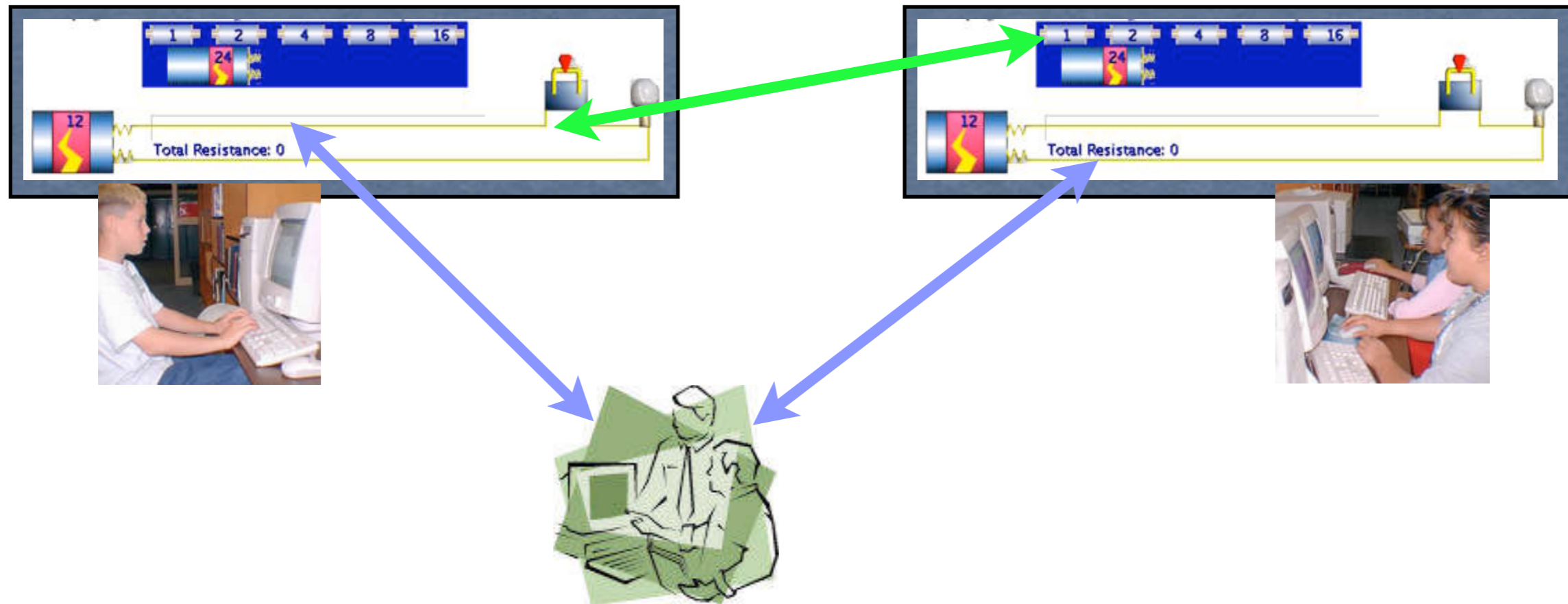
# What is it? (4)

## A moomie example

- example: a circuit simulator challenge:
- more than one student interacts at once
- programmatic objects 'talk to' each other
- ...so that students share the challenge
- tutors etc can 'listen' for key steps/actions
- authors can capture stages, help reflection

# What is it? (4)

## A moomie example



# What is it...(5) - example k-nase clientside services

- event-aware, knowledge-structuring chat
- reflective search tools
- shared text-editing
- report-generation / marking
- sundry specialist whiteboards / tools
- tutor support



# What is it... (6) - example k-nase remote services

- knowledge / multimedia-content repositories
- harvesting engines
- heavy-weight simulation engines
- metadata tagging / searching
- metadata/content exchange
- logging / tracking / synchronising...

# How will we develop it?

- It is a BIG job to do all of this 80)
- But...
  - It can be broken into stages
  - We will not do it alone
    - initially a 2-3 way collaboration
    - growing to an open-source community
- Go to standards organisations mid-way...

# Platforms to support?

- Server platforms:
  - FlashCom (first experiments)
  - Wep-app frameworks (eg Zope)
  - JAVA serverside...
- Client platforms:
  - Flash (first experiments)
  - html DOM/ecmascript
  - JAVA applets...

# Schedule of milestones

- Step 1 - feasibility, basic requirements, market research - is complete
- Step 2 - full-scale design
  - architecting/prototyping sprint May 10-17
- Step 3 - some functionality in-context
  - for open stories/platform-X by June 15
- Step 4 - major prototype by August 15
- Steps 5... - open-source collaborations

# Phase I - Open Stories

- As part of e-compete project, KnowNet and NewMediaLab are both committed to creating innovative learning activities from June 2004
- Open Stories is KnowNet's phase-I innovation for learning activities
  - it does not depend on k-nase/moomie
  - we are scheduling this in detail now
  - leverage existing apps like room-80
- We will also be assisting NewMediaLab in preparing similar half-way steps