e-Science Research Institute

National Research University of Information Technologies, Mechanics and Optics

Interactive workflow-based infrastructure for urgent computing

Workshop on Urgent Computing: Computations for Decision Support in Critical Situations (ICCS 2013)

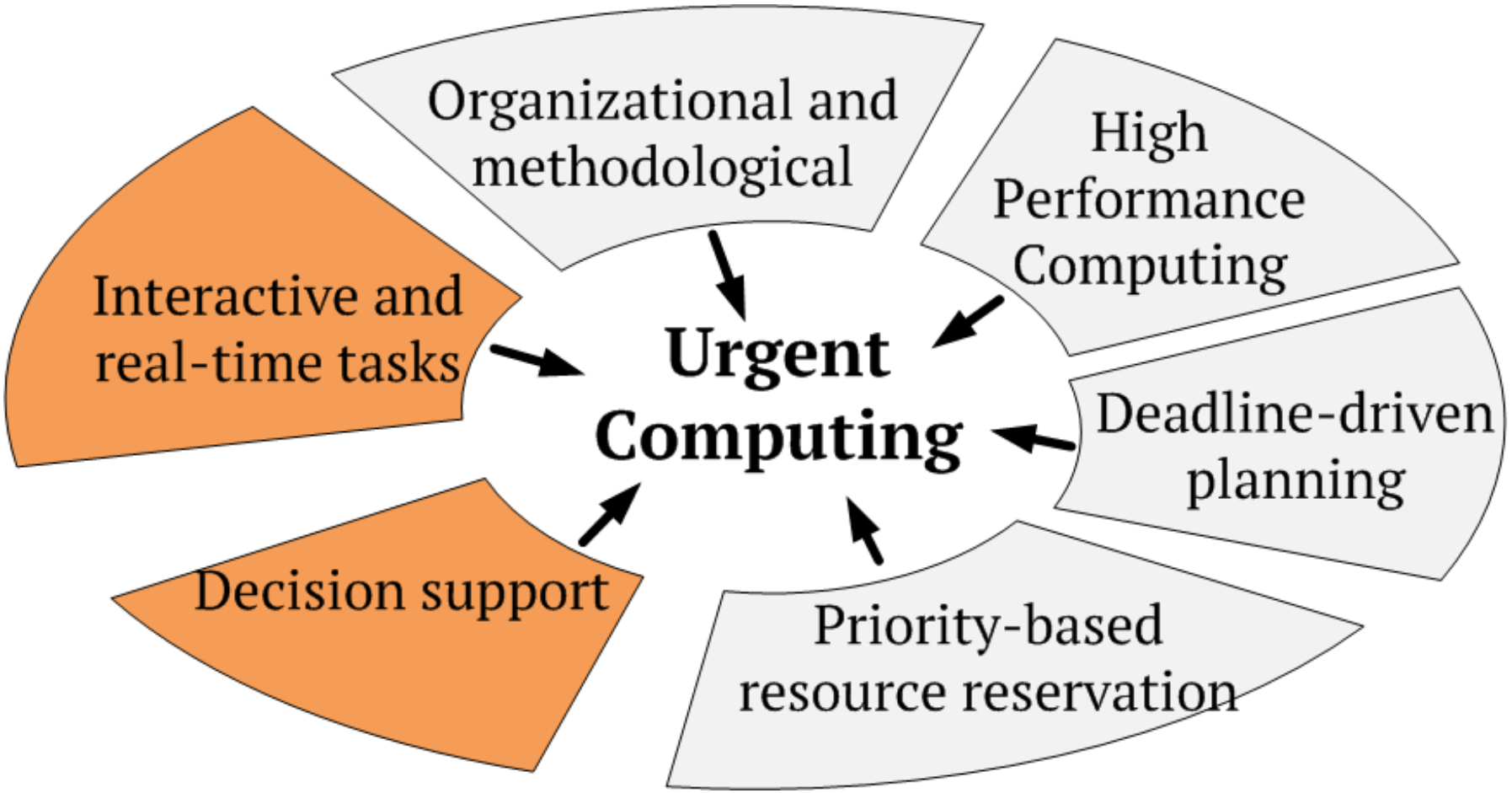
Barcelona / 07.06.2013 /

Konstantin Knyazkov

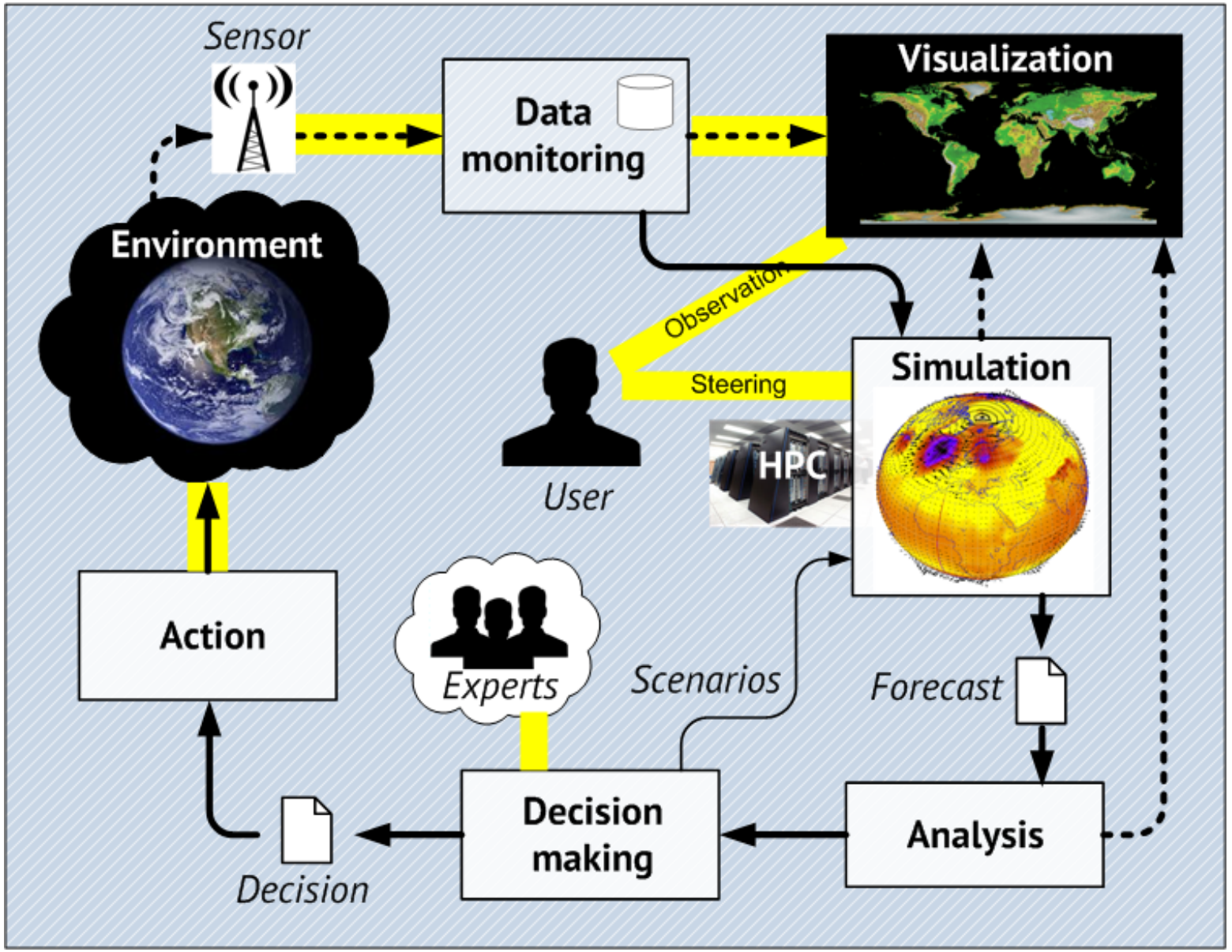
constantinvk@gmail.com

Denis Nasonov, Timofey Tchurov

Urgent Computing (UC) domain



Urgent Computing Infrastructure



Motivation

Prerequisites

1. Workflow became a mainstream formalism;
2. WMS are mostly batch and cannot provide interactive features;
3. UC domain requires complex workflows containing:
   - data acquisition,
   - data processing,
   - HPC support,
   - visualization,
   - real-time capabilities,
   - rich user interface.

➡

Goals

1. Provide convenient **UI** to simulation on a HPC resource;
2. Increase the **effectiveness** of resource usage by reduction of restart overhead;
3. Enable the capability of external interactive **devices** connection**.**

What do we need?

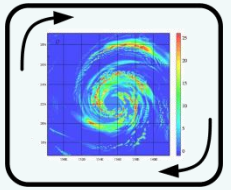
1. Abstraction and unification of interactive capabilities of software packages (from different domains) and their communication technologies;

2. Development technology for workflows based on interactive packages;

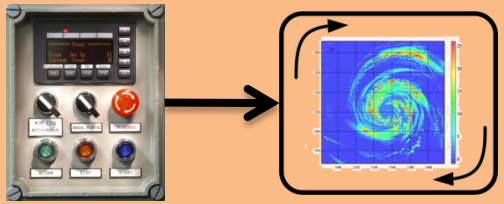
3. Technology implementation within a cyber infrastructure.

Interactive Workflow (IWF) principles

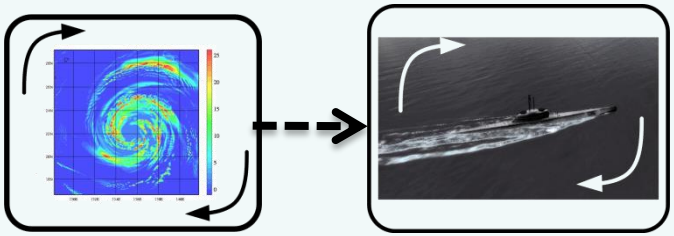
1. Blocks and workflow can be *long-running*;



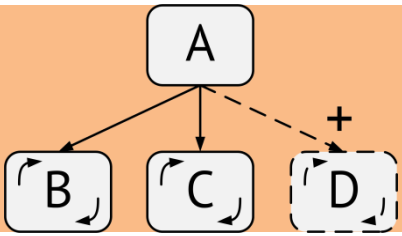
2. *Management* mechanisms for long-running block should be provided;



3. Blocks can *communicate* at runtime with each other and with an external environment;



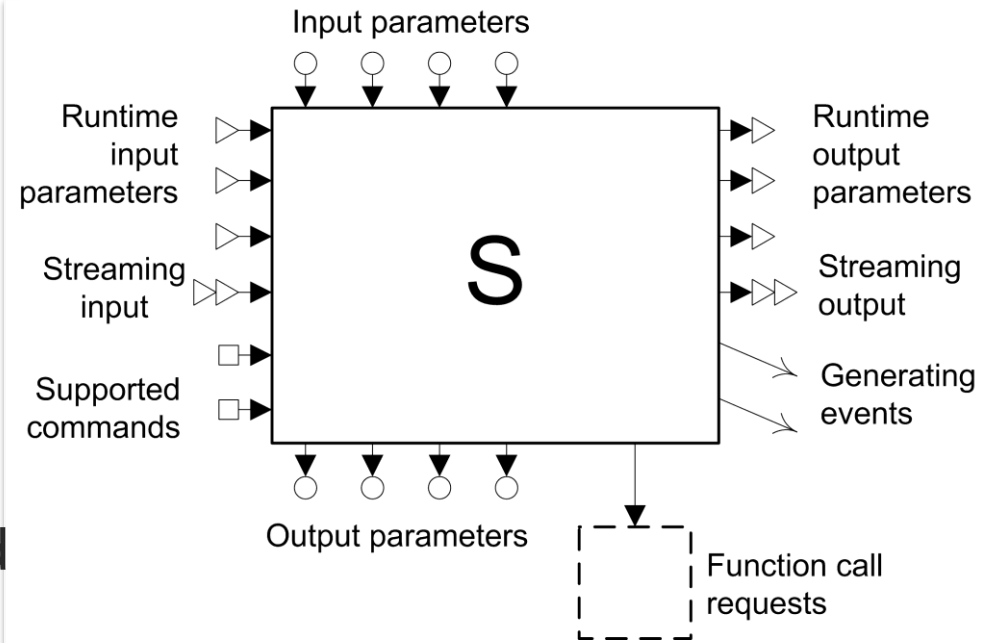
4. Capability of workflow's *modification* at runtime.

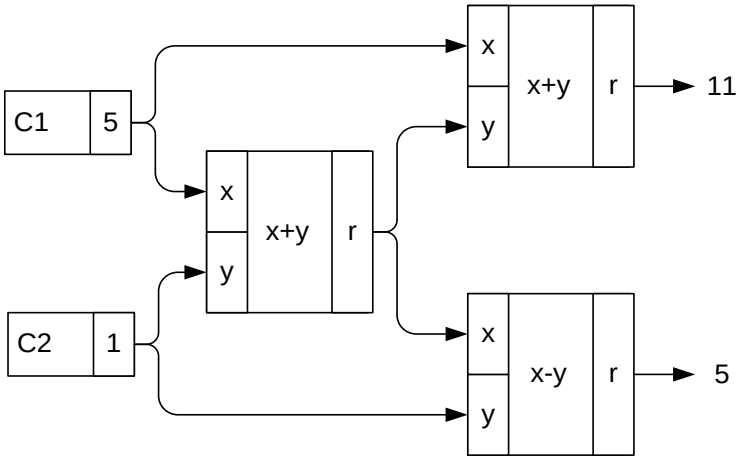


Interactive block model

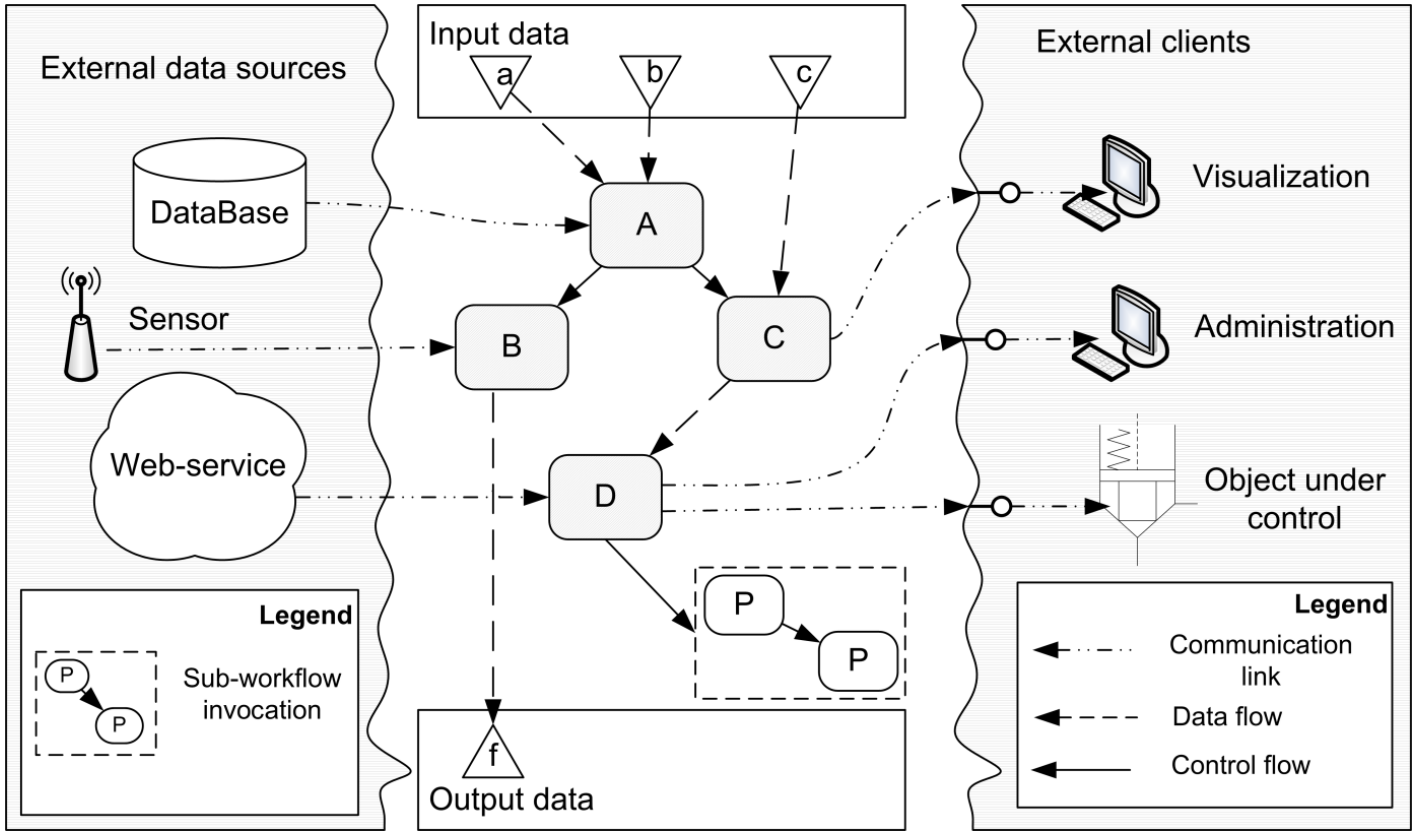
Block is a software package running on a resource and accessible through API.

- Communication
  - Runtime ports implementing **reactive programming** approach;
  - Generic ports characterized by protocols;
  - **Streaming** port for sending of unstructured data;
- Control
  - Commands;
  - Generating events;
  - Starting of sub-workflows.





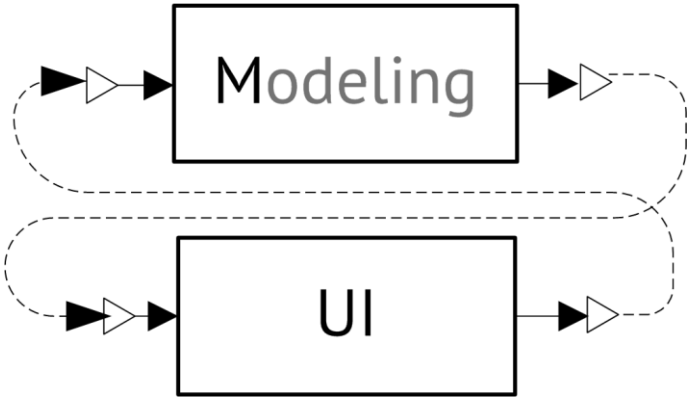
Interactive workflow model



1. Communication dependency connecting two ports;
2. Input and output workflow ports;
3. Control dependency now use events and commands.

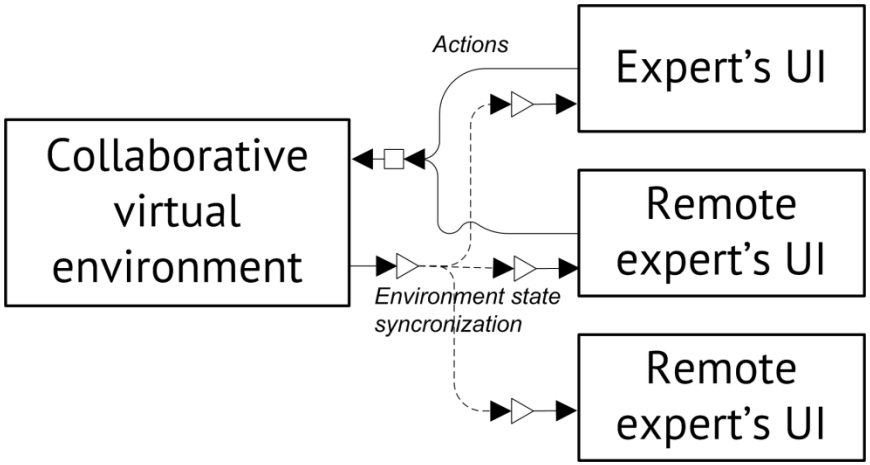
Interactive workflow patterns (1)

P1: Computational steering

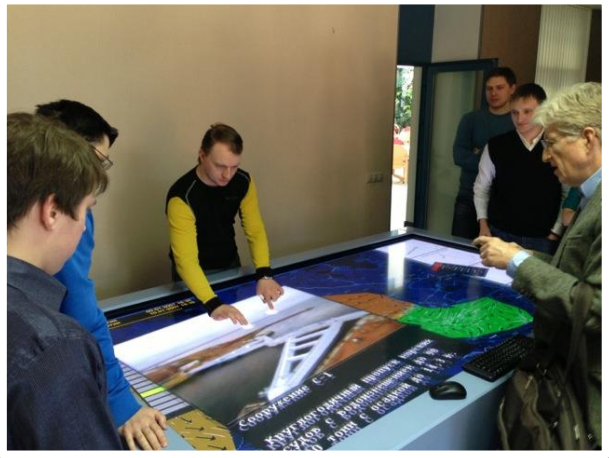


Example: Virtual environment for ship dynamics modeling

P2: Decision support tools

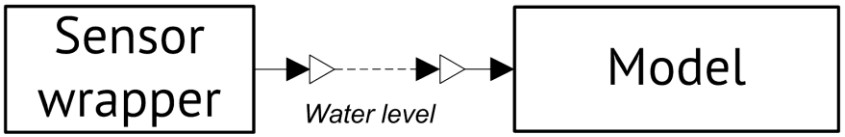


Example: Decision support system for dam control

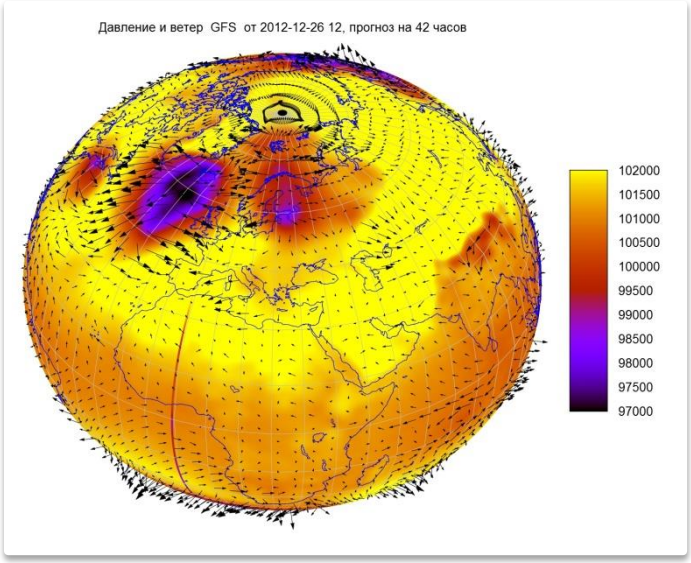


Interactive workflow patterns (2)

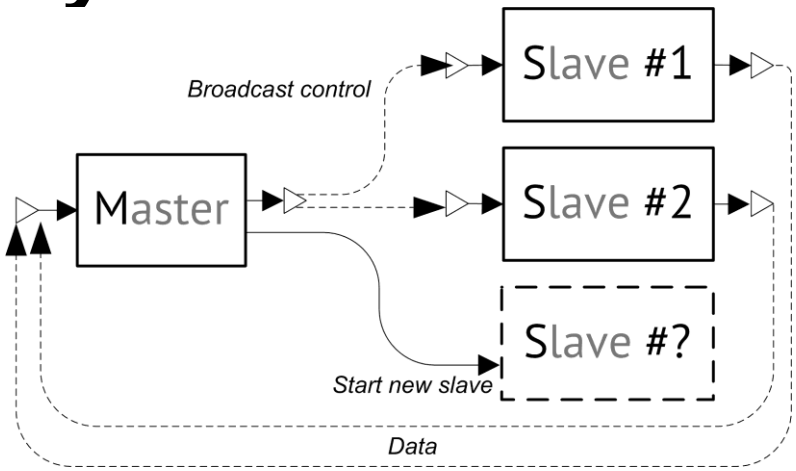
P3: Data assimilation



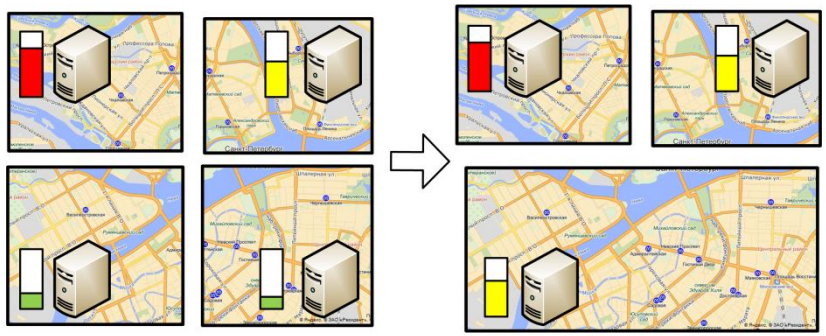
Example: Water level prediction model used by dam control center



P4: Dynamic load balancing



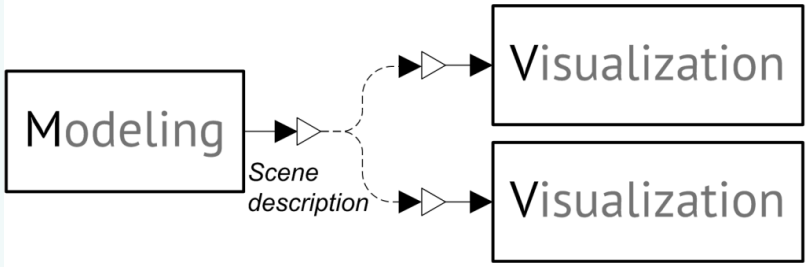
Example: Large-scale traffic multi agent modeling using distributed resources

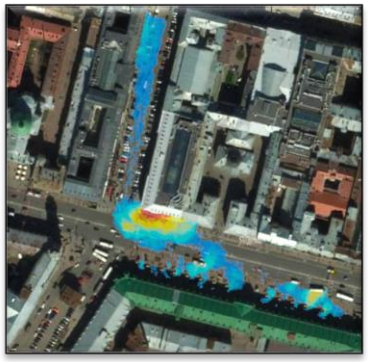


Interactive workflow patterns (3)

P5: Visualization and modeling separation

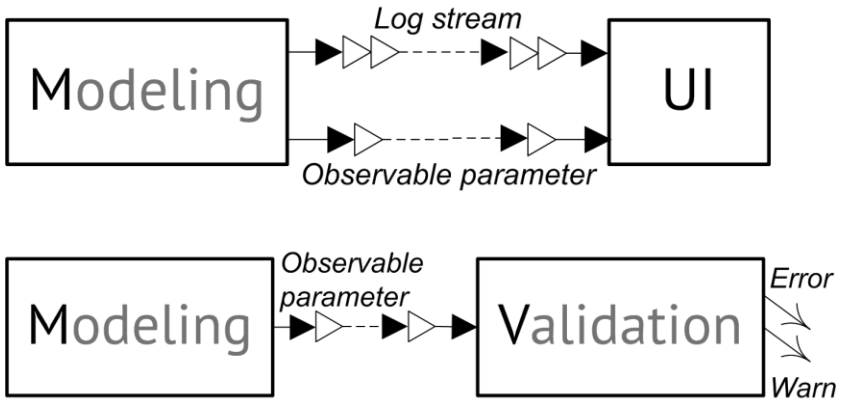
Example: Crowd management application

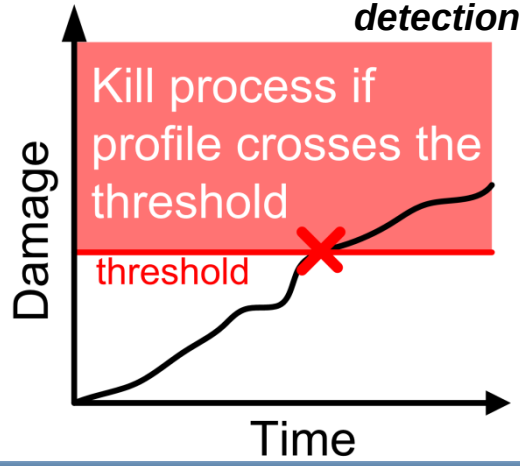




P6: Process observation and validation

Example: Runtime observation of the simulation processes in search problems for early detection of useless cases

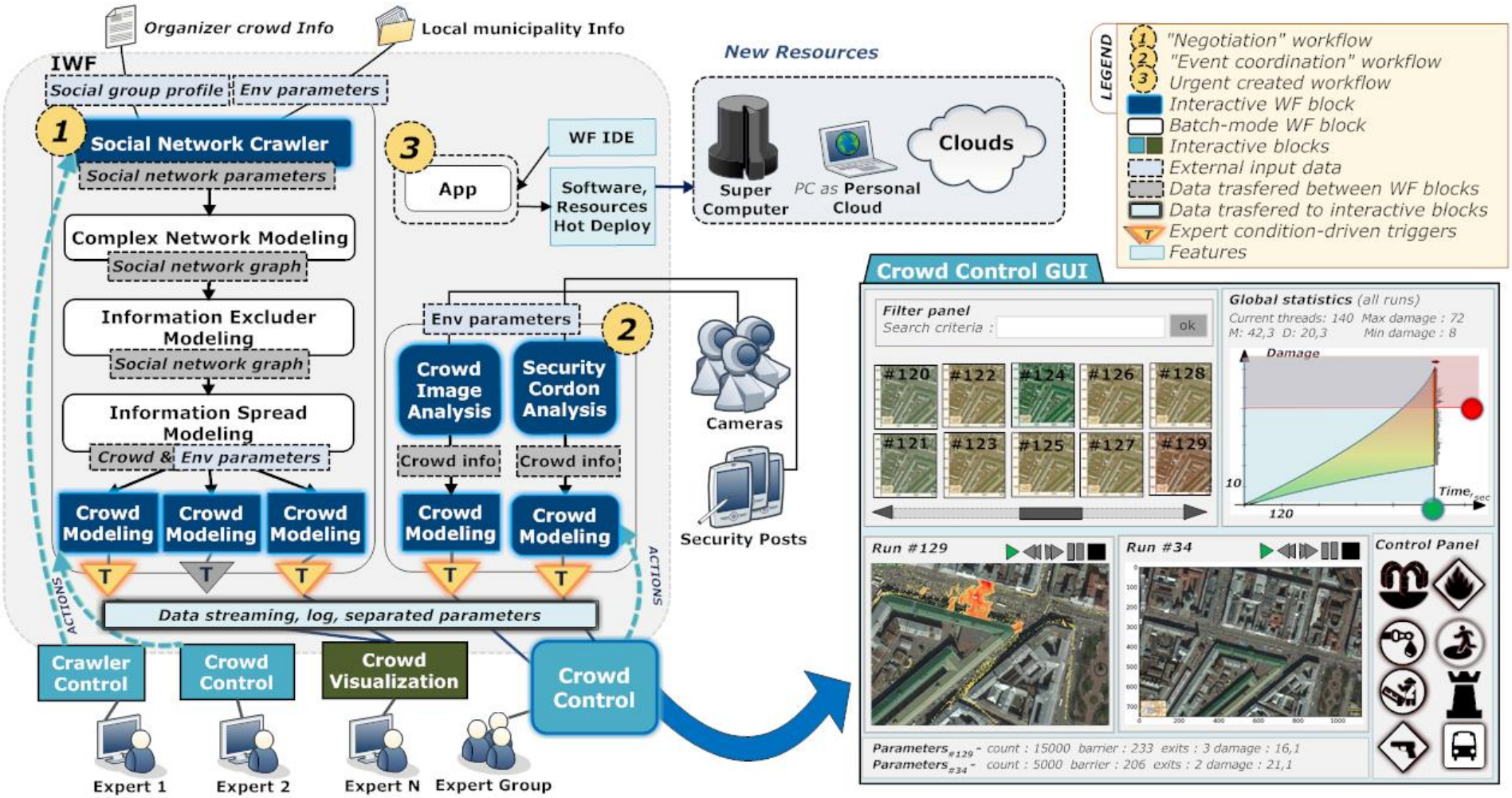




Adaptation of the CLAVIRE platform

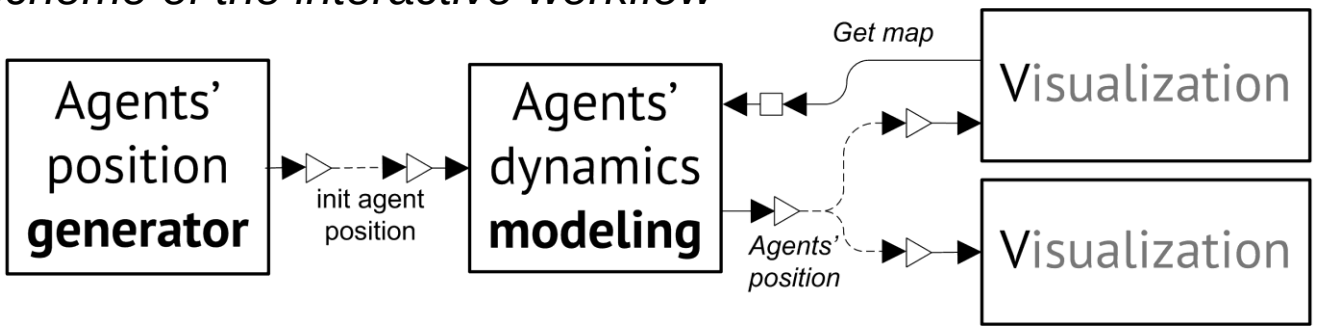
1. WMS ***CLAVIRE*** was taken as a base platform for experimenting;
2. ***Block description language*** was extended with port declaration capabilities;
3. ***Workflow description language*** *was* extended with port connection capabilities;
4. ***Communication dependencies*** were introduced in workflow management service;
5. ***Software library*** for interactive blocks implementing:
   - Communication environment using ZeroMQ and BSON;
   - Interactive block configuration mechanism using config files.

Use case: crowd management



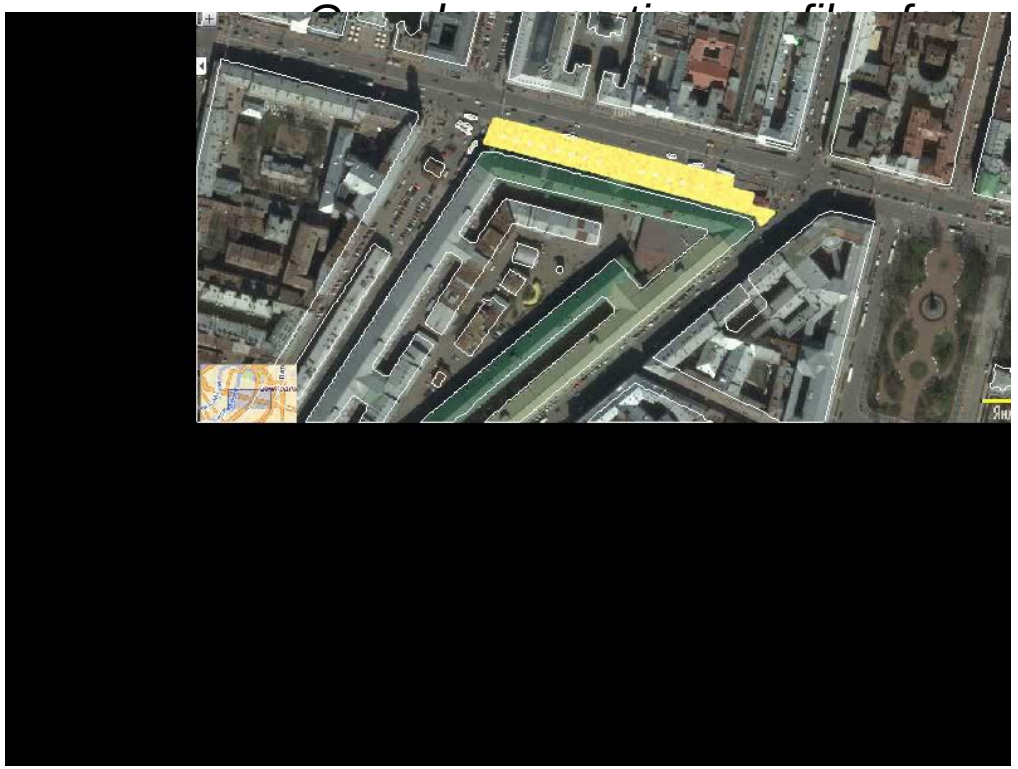
Implemented demonstration example

Scheme of the interactive workflow



Application consists of:

- Generator node updates agents positions after time interval (camera);
- Evacuation dynamics modeling block simulates crowd flow;
- Several visualization blocks show the same picture of running crowd;
- Capability to define different map settings.

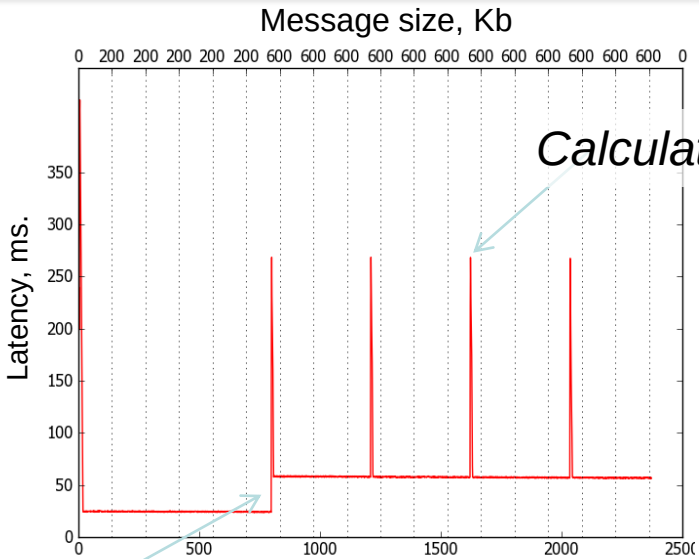


Model time, ticks

Experiment results

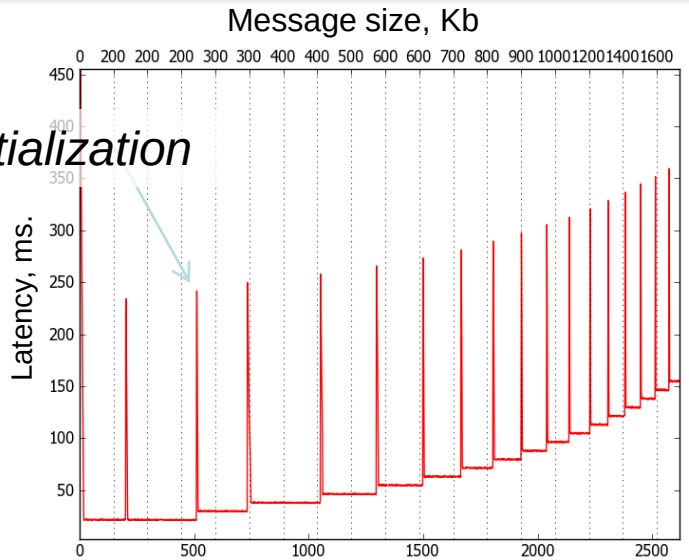
Constant size of crowd

Latency



Calculation initialization

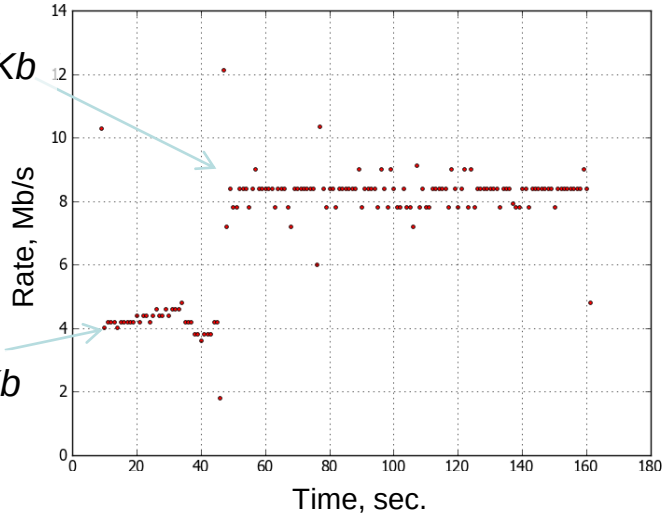
Crowd size changing



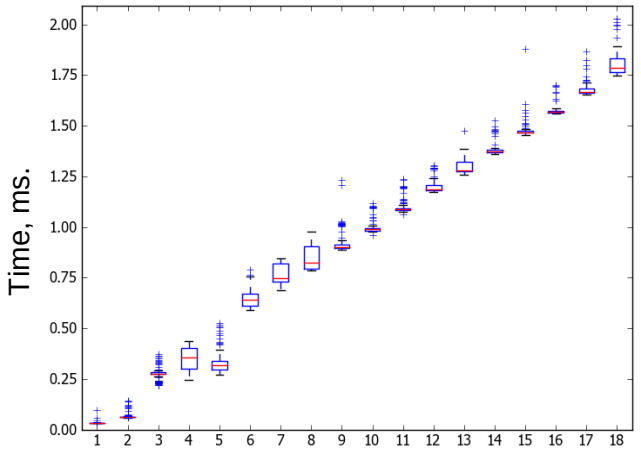
Rate of data transmition to visualization process

600 Kb

200 Kb



Data pack time related to message size

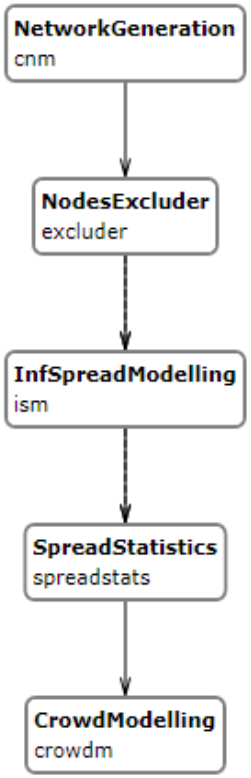


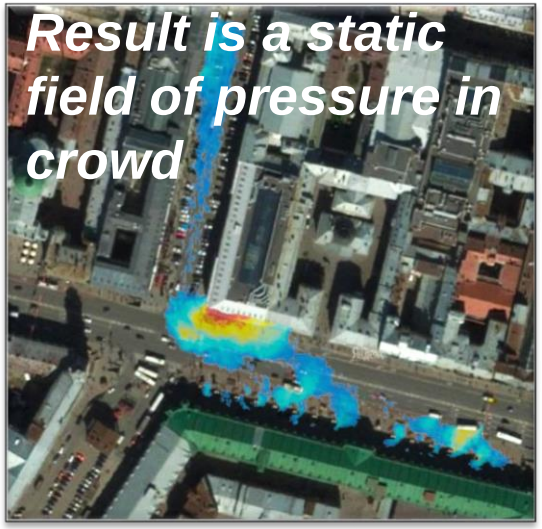
Flash mob evacuation modeling workflow

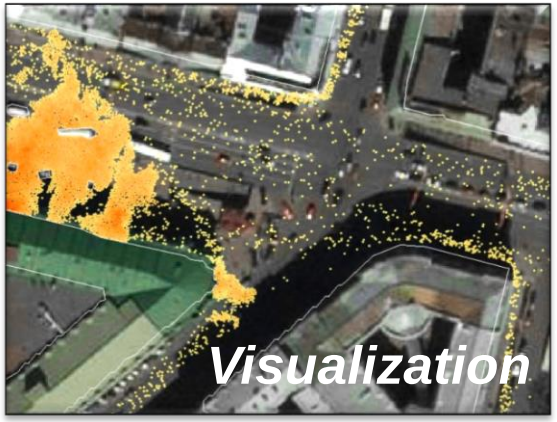
Demonstration application for studying of critical situations occurred during unauthorized events which were organized through social networks (crowd panic modeling)

Application contains stages:

1. Complex network generation;
2. Random nodes excluding;
3. Information spread modeling through the group of networks;
4. Aggregation and calculation of statistic parameters;
5. Evacuation dynamic modeling with interactive visualization;







Conclusion

- The ***main idea*** is to consider interactivity as a part of UC and to consider it as a part of WF abstraction;
- The proposed model and implementation showed its ***feasibility*** in our use case:
  - allowed visualization and modeling separation;
  - allowed computational steering (change map at runtime);
- The prototype provides enough network performance for further research;
- ***Future*** research contains: full use case implementation; network-related problems (bandwidth); integrating heterogeneous resources.