Grid programming with components: an advanced COMPonent platform for an effective invisible grid

GridCOMP
Autonomic Features
22 Sep 2008 - Bruxelles

UNIPI & ISTI-CNR
Dicom demo: rationale

- What it is?
  - X-ray images analysis (mammographic density)
- Which GridCOMP WP3 results does it exploit?
  - Stateless, stream-based functional replication
  - Active “farm” behavioural skeleton
  - Self-optimising autonomic features
- What you should expect?
  - N. of workers is dynamically adjusted with w.r.t. the actual platform status and application behaviour
- Modus operandi
  - Platform load is artificially changed during the run
Dicom demo: screen output

- Image detail
- Thumbnails of processed images
- Manager activity
  - Monitor & reaction
Dicom demo: “farm” behavioural skeleton
Dicom demo: dynamic recruiting of resources
Dicom demo: rationale

- What it is?
  - X-ray images images analysis (mammographic density)

- Which GridCOMP WP3 results does it exploit?
  - Stateless, stream-based functional replication
  - Active “farm” behavioural skeleton
  - Self-optimising autonomic features

- What you should expect?
  - N. of workers is dynamically adjusted with w.r.t. the actual platform status and application behaviour

- Modus operandi
  - Platform load is artificially changed during the run
IBM app mockup demo: rationale

- What it is?
  - **IBM fingerprint app mockup**
- Which GridCOMP WP3 results does it exploit?
  - Stateful, service-based functional replication
  - Active “data-parallel” behavioural skeleton
  - Self-configuring autonomic features
- What you should expect?
  - **N. of workers is dynamically adjusted with w.r.t. the dataset size to keep almost constant data-per-worker size (i.e. memory footprint and service execution time)**
- Modus operandi
  - **Dataset size is changed during the run**
IBM app mockup demo: screen output

“fingerprint” matching results

manager activity monitor & reaction
IBM app mockup: stateful “DP” behavioural skeleton
Mandelbrot example (two-level management)

Grid programming with components: an advanced COMPonent platform for an effective invisible grid
IBM app mockup demo: rationale

What it is?

- **IBM fingerprint app mockup**

Which GridCOMP WP3 results does it exploit?

- **Stateful, service-based functional replication**
- **Active “data-parallel” behavioural skeleton**
- **Self-configuring autonomic features**

What you should expect?

- **N. of workers is dynamically adjusted with w.r.t. the dataset size to keep almost constant data-per- worker size (i.e. memory footprint and service execution time)**

Modus operandi

- **Dataset size is changed during the run**
IBM app mockup: contract (JBoss drools)

rule "CheckHigherBound"
  when
    $arrivalBean : PartitionSizeBean(value >= 70)
    $count : CounterBean( value > 2 )
  then
    $arrivalBean.fireOperation(ManagerOperation.ADD_EXECUTOR);
    $count.reset(); retract($arrivalBean); update( $count );
end

rule "CheckLowerBound"
  when
    $arrivalBean : PartitionSizeBean(value <= 50)
    $count : CounterBean( value > 2 )
  then
    $arrivalBean.fireOperation(ManagerOperation.REMOVE_EXECUTOR);
    $count.reset(); retract($arrivalBean); update( $count );
end

rule "Count"
  salience -10
  when
    $arrivalBean : PartitionSizeBean( )
    $count : CounterBean( value <= 2 )
  then
    $count.incValue(); retract($arrivalBean); update( $count );
end