EVEREST+
RUN-TIME SLA VIOLATION PREDICTION

Davide Lorenzoli, George Spanoudakis

MW4SOC, Bangalore, 29/11/2010
Outline

• Prediction approaches limitations
• Our vision
• EVEREST+
• Experimental results
• Conclusions
Prediction approaches limitations

• They tend to focus on system infrastructure properties rather than service level application based properties.

• They tend to focus on the prediction of specific types of properties without providing a more generic framework for building predictors.

• They are not integrated with environments for monitoring SLAs for service-based systems.
Our vision

- To focus on system infrastructure properties and service level application based properties.
- To provide a more generic framework for building predictors that can cover a wide or even the whole spectrum of service properties that can be part of an SLA.
- To integrate with environments for monitoring SLAs for service-based systems.
EVEREST+

• EVEREST+ is a framework for integrated monitoring and prediction

• EVEREST+ uses prediction specifications to setup both the monitoring and the prediction framework

• EVEREST+ provides the means for developing new predictors
EVEREST+: architecture

**EVEREST+**

**Prediction Framework**

- **QoS predictor**
- **Model**
- **Model manager**
- **Prediction**
- **Monitoring specificat. generator**

**EVEREST**

- **QoS**
- **QoS monitor**
- **QoS specification**
- **SLA violation**

**Prediction specificat.**
- Agreement term
- Prediction parameters
- Predictor configurat.
- QoS specification

**read**
**write**
**communication**
EVEREST+: architecture

- Agreement term
- Prediction parameters
- Predictor configuration
- QoS specification

**Prediction Framework**

**Monitoring specification generator**

**read**

**write**

**communication**
EVEREST+: architecture

**Prediction Framework**

**EVEREST**

- Agreement term
- Prediction parameters
- Predictor configurat.
- QoS specification
EVERST+: monitoring framework EVERST

- Generic

- Based on Event Calculus (EC)

- Rules:
  - body $\Rightarrow$ head

- Predicates:
  - $\text{Happens}(e,t,R(1b,ub))$
  - $\text{HoldsAt}(f,t)$
  - $\text{Initiates}(e,f,t)$
  - $\text{Terminates}(e,f,t)$
  - $\text{Initially}(f)$
EVERST+: monitoring framework EVERST

• Mean Time To Repair (MTTR) QoS: the formula checks whether the MTTR of service \(_Srv\) is always below a given threshold \(K\), i.e., \(MTTR<K\).

Rule R1:

\[\text{Happens}(e(_id1, _Snd, _Srv, \text{Call}(O), _Srv), t_1, [t_1, t_1]) \land \]
\[\text{Happens}(e(_id2, _Srv, _Snd, \text{Response}(O), _Srv), t_2, [t_1, t_1+d]) \land \]
\[\exists _{PN}, _{STime}, _{MTTR}: \text{HoldsAt}(\text{Unavailable}(_{PN}, _{Srv}, _{STime}), t_1)) \land \]
\[\text{HoldsAt}(MTTR(_{Srv}, _{PN}, _{MTTR}), t_1)) \]
\[\Rightarrow _{MTTR} < K\]
EVERST+: monitoring framework EVERST

- Mean Time To Repair (MTTR) QoS: the formula checks whether the MTTR of service \(_Srv\) is always below a given threshold \(K\), i.e., MTTR<\(K\).

**Rule R1:**

\[
\text{Happens}(e(_id1, _Snd, _Srv, \text{Call(_O), _Srv}), t_1, [t_1, t_1]) \land \\
\text{Happens}(e(_id2, _Srv, _Snd, \text{Response(_O), _Srv}), t_2, [t_1, t_1+d]) \land \\
\exists _PN, _STime, _MTTR: \text{HoldsAt(Unavailable(_PN, _Srv, _STime), t_1)}) \land \\
\text{HoldsAt(MTTR(_Srv, _PN, _MTTR), t_1)}) \\
\Rightarrow _\text{MTTR} < K
\]
EVERST+: monitoring framework EVERST

- Mean Time To Repair (MTTR) QoS: the formula checks whether the MTTR of service \( _Srv \) is always below a given threshold \( K \), i.e., \( \text{MTTR} < K \).

**Rule R1:**

\[
\text{Happens}(e(_id1, _Snd, _Srv, \text{Call}(O), _Srv), t_1, [t_1, t_1]) \land \\
\text{Happens}(e(_id2, _Srv, _Snd, \text{Response}(O), _Srv), t_2, [t_1, t_1+d]) \land \\
\exists _PN, _STime, _MTTR: \text{HoldsAt}(\text{Unavailable}(_PN, _Srv, _STime), t_1)) \land \\
\text{HoldsAt}(\text{MTTR}(_Srv, _PN, _MTTR), t_1))
\]

\[\Rightarrow _MTTR < K\]

A response from service \( _Srv \) is received at time point \( t_1+d \).
EVERST+: monitoring framework EVERST

- Mean Time To Repair (MTTR) QoS: the formula checks whether the MTTR of service $Srv$ is always below a given threshold $K$, i.e., $MTTR < K$.

Rule R1:

$\text{Happens}(e(_id1, _Snd, _Srv, \text{Call}(O), _Srv), t_1) \wedge \text{Happens}(e(_id2, _Srv, _Snd, \text{Response}(O), _Srv), t_2, [t_1, t_1 + d]) \wedge$

$\exists _PN, _STime, _MTTR: \text{HoldsAt}(\text{Unavailable}(_PN, _Srv, _STime), t_1)) \wedge \text{HoldsAt}(MTTR(_Srv, _PN, _MTTR), t_1))$

$\Rightarrow _MTTR < K$
EVERST+: monitoring framework EVERST

- Mean Time To Repair (MTTR) QoS: the formula checks whether the MTTR of service \(_Srv\) is always below a given threshold \(K\), i.e., \(MTTR < K\).

**Rule R1:**

\[
\text{Happens}(e(_id1, _Snd, _Srv, \text{Call}(\_O), _Srv), t_1, [t_1, t_1]) \land \text{Happens}(e(_id2, _Srv, _Snd, \text{Response}(\_O), _Srv), t_2, [t_1, t_1 + d]) \land \exists _PN, _STime, _MTTR: \text{HoldsAt}(\text{Unavailable}(_PN, _Srv, _STime), t_1)) \land \text{HoldsAt}(MTTR(_Srv, _PN, _MTTR), t_1)) \\
\Rightarrow _MTTR < K
\]

Checks for MTTR violations \((MTTR \geq K)\) when a call to an operation \(_O\) of the service \(_Srv\) is served after a period of unavailability.
EVERST+: monitoring framework EVERST

- After receiving a monitoring specification
  - computes/store MTTR values
  - checks for MTTR violations
EVEREST+: prediction framework

Prediction Framework

Predictor

Model

Model manager

Prediction

Monitoring specificat. generator

EC monitoring specification

Prediction manager

QoS predictor

Prediction specificat.
- Agreement term
- Prediction parameters
- Predictor configurat.
- QoS specification

read
write
communication
EVEREST+: specification driven

- Agreement term
- Prediction parameters
- Predictor configuration
- QoS specification
EVEREST+: specification driven

- MTTR < K
- \( t_c = <\text{now}> \)
- \( p = 1000s \)
- QoS={MTTR, MTTF}

Automatically inferred models

- Activates MTTR and MTTF model generator
- Forwards specification
- Event Calculus specification for monitoring MTTR and MTTF

QoS monitor

Model manager

Model

MTTR predictor

Prediction manager

Prediction specificat.
EVEREST+: generic framework for building predictors
EVEREST+: generic framework for building predictors
Prediction problem

Pr(QoS, K, t_e): Given a request for predicting whether a QoS property will violate a given constraint K set for it at some future time point t_e that is received at a time point t_c, prediction is the computation of the probability that the QoS property will violate the constraint at t_e.
EVEREST+: generic framework for building predictors
EVEREST+: generic framework for building predictors

\[ \Pr(\bigwedge_{y=1}^{M} E_y) = \begin{cases} 1 - \sum_{y=1}^{M} [\Pr(y) \cdot \Pr(MTTR_{\text{crit}} > MTTR_y)], & MTTR_{\text{crit}} > K \\ \sum_{y=1}^{M} [\Pr(y) \cdot \Pr(MTTR_{\text{crit}} \leq MTTR_y)], & MTTR_{\text{crit}} \leq K \end{cases} \]
EVEREST+: generic framework for building predictors

\[ \Pr\left( E_{y} \right) = \begin{cases} 
1 - \sum_{y=1}^{M} \left[ \Pr(y) \times \Pr(\text{MTTR}_{\text{crit}} > \text{MTTR}_{y}) \right], & \text{MTTR}_{\text{crit}} > K \\
\sum_{y=1}^{M} \left[ \Pr(y) \times \Pr(\text{MTTR}_{\text{crit}} \leq \text{MTTR}_{y}) \right], & \text{MTTR}_{\text{crit}} \leq K 
\end{cases} \]
### Experimental results

- **4 MTTR Trends:**
  - *T1: cyclic*
  - *T2: increasing*
  - *T3: decreasing*
  - *T4: random*

- **3 Variables:**
  - History size
  - Prediction window
  - Goodness of fit

- **40 prediction measures**

<table>
<thead>
<tr>
<th></th>
<th>T1 P</th>
<th>T1 R</th>
<th>T2 P</th>
<th>T2 R</th>
<th>T3 P</th>
<th>T3 R</th>
<th>T4 P</th>
<th>T4 R</th>
<th>Overall P</th>
<th>Overall R</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>HS size</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>100 events</td>
<td>.78</td>
<td>.98</td>
<td>.77</td>
<td>.69</td>
<td>.76</td>
<td>1.0</td>
<td>.81</td>
<td>.77</td>
<td>.78</td>
<td>.80</td>
</tr>
<tr>
<td>300 events</td>
<td>.78</td>
<td>.98</td>
<td>.77</td>
<td>.69</td>
<td>.76</td>
<td>1.0</td>
<td>.81</td>
<td>.77</td>
<td>.78</td>
<td>.80</td>
</tr>
<tr>
<td>500 events</td>
<td>.78</td>
<td>.98</td>
<td>.77</td>
<td>.69</td>
<td>.76</td>
<td>1.0</td>
<td>.81</td>
<td>.76</td>
<td>.78</td>
<td>.80</td>
</tr>
<tr>
<td><strong>PW length</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1sec</td>
<td>.66</td>
<td>.91</td>
<td>.57</td>
<td>.54</td>
<td>.52</td>
<td>1.0</td>
<td>.68</td>
<td>.63</td>
<td>.61</td>
<td>.62</td>
</tr>
<tr>
<td>1min</td>
<td>.70</td>
<td>1.0</td>
<td>.75</td>
<td>.67</td>
<td>.75</td>
<td>1.0</td>
<td>.76</td>
<td>.73</td>
<td>.74</td>
<td>.77</td>
</tr>
<tr>
<td>10mins</td>
<td>.66</td>
<td>.91</td>
<td>.57</td>
<td>.54</td>
<td>.52</td>
<td>1.0</td>
<td>.68</td>
<td>.63</td>
<td>.61</td>
<td>.62</td>
</tr>
<tr>
<td><strong>GoF</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[.0-.05]</td>
<td>.82</td>
<td>.96</td>
<td>.80</td>
<td>.71</td>
<td>.76</td>
<td>1.0</td>
<td>.83</td>
<td>.75</td>
<td>.78</td>
<td>.85</td>
</tr>
<tr>
<td>(.05-.1]</td>
<td>.78</td>
<td>.99</td>
<td>.77</td>
<td>.68</td>
<td>.75</td>
<td>1.0</td>
<td>.81</td>
<td>.78</td>
<td>.78</td>
<td>.80</td>
</tr>
<tr>
<td>(.1-.15]</td>
<td>.74</td>
<td>1.0</td>
<td>.75</td>
<td>.67</td>
<td>n/a</td>
<td>n/a</td>
<td>.81</td>
<td>.75</td>
<td>.79</td>
<td>.75</td>
</tr>
<tr>
<td>(.15-.2]</td>
<td>.72</td>
<td>1.0</td>
<td>n/a</td>
<td>n/a</td>
<td>n/a</td>
<td>n/a</td>
<td>.82</td>
<td>.76</td>
<td>.80</td>
<td>.78</td>
</tr>
</tbody>
</table>
### Experimental results

- **4 MTTR Trends:**
  - *T1*: cyclic
  - *T2*: increasing
  - *T3*: decreasing
  - *T4*: random

- **3 Variables:**
  - History size
  - Prediction window
  - Goodness of fit

- **40 prediction measures**

<table>
<thead>
<tr>
<th></th>
<th>T1</th>
<th>T2</th>
<th>T3</th>
<th>T4</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P</td>
<td>R</td>
<td>P</td>
<td>R</td>
<td>P</td>
</tr>
<tr>
<td><strong>HS size</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>100 events</td>
<td>.78</td>
<td>.98</td>
<td>.77</td>
<td>.69</td>
<td>.76</td>
</tr>
<tr>
<td>300 events</td>
<td>.78</td>
<td>.98</td>
<td>.77</td>
<td>.69</td>
<td>.76</td>
</tr>
<tr>
<td>500 events</td>
<td>.78</td>
<td>.98</td>
<td>.77</td>
<td>.69</td>
<td>.76</td>
</tr>
<tr>
<td><strong>PW length</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 sec</td>
<td>.66</td>
<td>.91</td>
<td>.57</td>
<td>.54</td>
<td>.52</td>
</tr>
<tr>
<td>1 min</td>
<td>.70</td>
<td>1.0</td>
<td>.75</td>
<td>.67</td>
<td>.75</td>
</tr>
<tr>
<td>10 mins</td>
<td>.66</td>
<td>.91</td>
<td>.57</td>
<td>.54</td>
<td>.52</td>
</tr>
<tr>
<td><strong>GoF</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[.0-.05]</td>
<td>.82</td>
<td>.96</td>
<td>.80</td>
<td>.71</td>
<td>.76</td>
</tr>
<tr>
<td>(.05-.1]</td>
<td>.78</td>
<td>.99</td>
<td>.77</td>
<td>.68</td>
<td>.75</td>
</tr>
<tr>
<td>(.1-.15]</td>
<td>.74</td>
<td>1.0</td>
<td>.75</td>
<td>.67</td>
<td>n/a</td>
</tr>
<tr>
<td>(.15-.2]</td>
<td>.72</td>
<td>1.0</td>
<td>n/a</td>
<td>n/a</td>
<td>n/a</td>
</tr>
</tbody>
</table>
Experimental results

- 4 MTTR Trends:
  - *T1*: cyclic
  - *T2*: increasing
  - *T3*: decreasing
  - *T4*: random

- 3 Variables:
  - History size
  - Prediction window
  - Goodness of fit

- 40 prediction measures

<table>
<thead>
<tr>
<th></th>
<th>T1</th>
<th>T2</th>
<th>T3</th>
<th>T4</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>#predictions</strong></td>
<td>1440</td>
<td>1440</td>
<td>1440</td>
<td>1440</td>
<td>4320</td>
</tr>
<tr>
<td><strong>precision</strong></td>
<td>0.78</td>
<td>0.77</td>
<td>0.76</td>
<td>0.81</td>
<td>.78</td>
</tr>
<tr>
<td><strong>recall</strong></td>
<td>0.98</td>
<td>0.69</td>
<td>1.00</td>
<td>0.77</td>
<td>.80</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>T1</th>
<th>T2</th>
<th>T3</th>
<th>T4</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>HS size</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>100 events</td>
<td>.78</td>
<td>.98</td>
<td>.77</td>
<td>.69</td>
<td>.76</td>
</tr>
<tr>
<td>300 events</td>
<td>.78</td>
<td>.98</td>
<td>.77</td>
<td>.69</td>
<td>.76</td>
</tr>
<tr>
<td>500 events</td>
<td>.78</td>
<td>.98</td>
<td>.77</td>
<td>.69</td>
<td>.76</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>T1</th>
<th>T2</th>
<th>T3</th>
<th>T4</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PW length</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1sec</td>
<td>.66</td>
<td>.91</td>
<td>.57</td>
<td>.54</td>
<td>.52</td>
</tr>
<tr>
<td>1min</td>
<td>.70</td>
<td>1.0</td>
<td>.75</td>
<td>.67</td>
<td>.75</td>
</tr>
<tr>
<td>10mins</td>
<td>.66</td>
<td>.91</td>
<td>.57</td>
<td>.54</td>
<td>.52</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>T1</th>
<th>T2</th>
<th>T3</th>
<th>T4</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>GoF</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[.0-.05]</td>
<td>.82</td>
<td>.96</td>
<td>.80</td>
<td>.71</td>
<td>.76</td>
</tr>
<tr>
<td>(.05-.1]</td>
<td>.78</td>
<td>.99</td>
<td>.77</td>
<td>.68</td>
<td>.75</td>
</tr>
<tr>
<td>(.1-.15]</td>
<td>.74</td>
<td>1.0</td>
<td>.75</td>
<td>.67</td>
<td>n/a</td>
</tr>
<tr>
<td>(.15-.2]</td>
<td>.72</td>
<td>1.0</td>
<td>n/a</td>
<td>n/a</td>
<td>n/a</td>
</tr>
</tbody>
</table>
Conclusions & Future Work

• EVEREST+ is a framework for integrated monitoring and prediction

• EVEREST+ uses prediction specifications to setup both the monitoring and the prediction framework

• EVEREST+ provides the means for developing new predictors

• Testing existing predictors against data coming from different contexts

• Designing and implementation of a wider set of predictors

• Everest+ support for other monitoring frameworks
Bibliography


THANK YOU