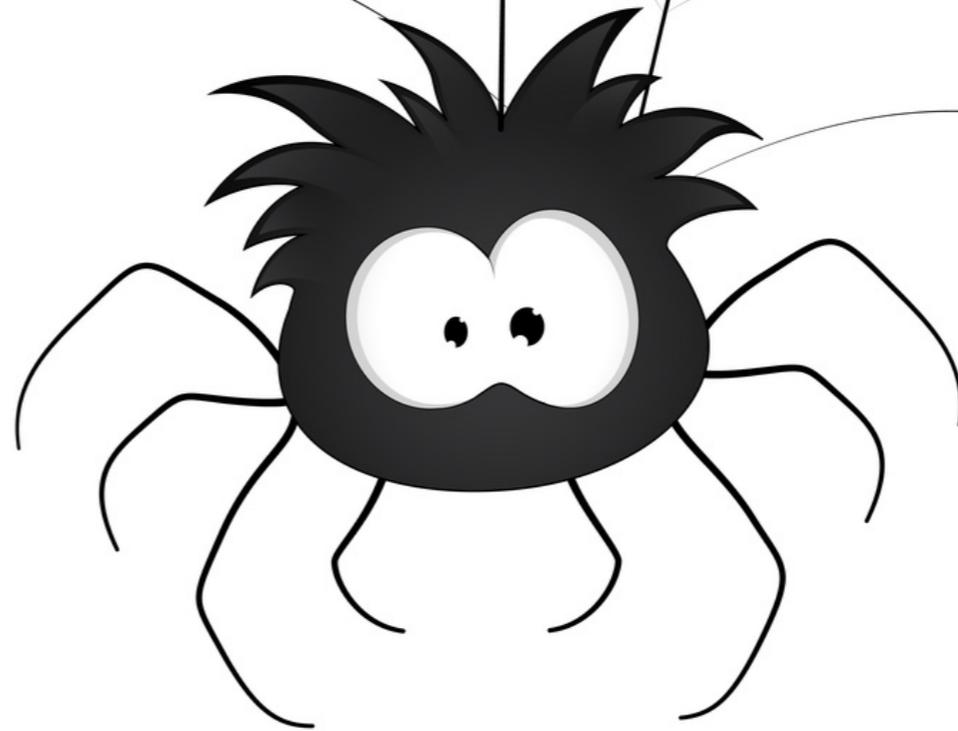


# Xtext IN THE WEB



**Who s that guy?**



**Why in the Web???**



# BUSINESS DSLS

- Designed for business experts
- Formalize the business aspects of an application
- Examples
  - business rules
  - workflows
  - role models
  - application configuration
- Usually executed at runtime (not design time)

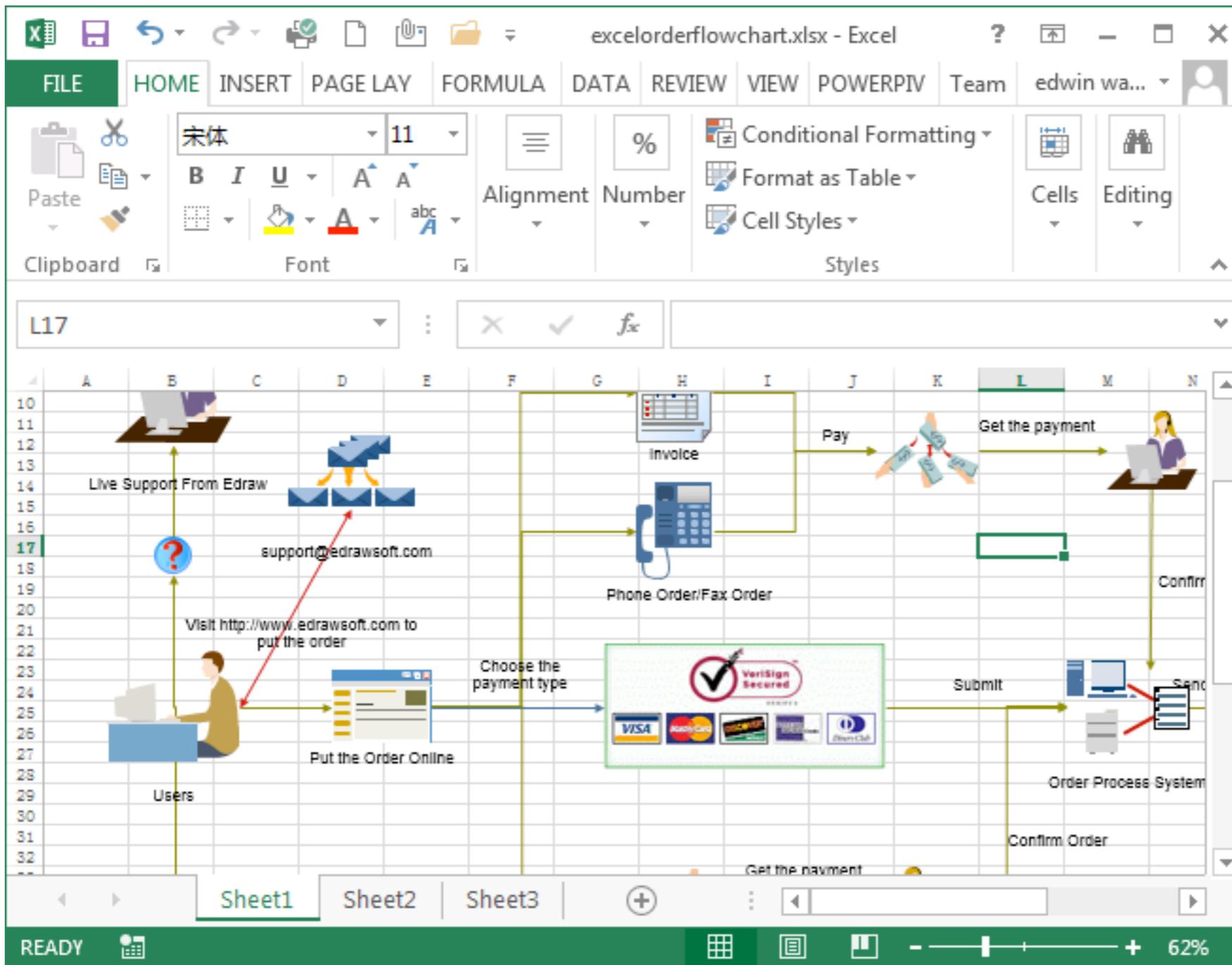


# SOFTWARE DEVELOPER VS BUSINESS USER

- Software Developer
  - Trained to use IDEs
  - Loves text



- Business User
  - Excel
  - Web-based tools
  - Wants a GUI



NOT EVERYTHING IS A TABLE...

**Guided Rule Editor [No bad credit checks]** Save Delete Rename Copy Validate X

**EXTENDS** None selected

**WHEN**

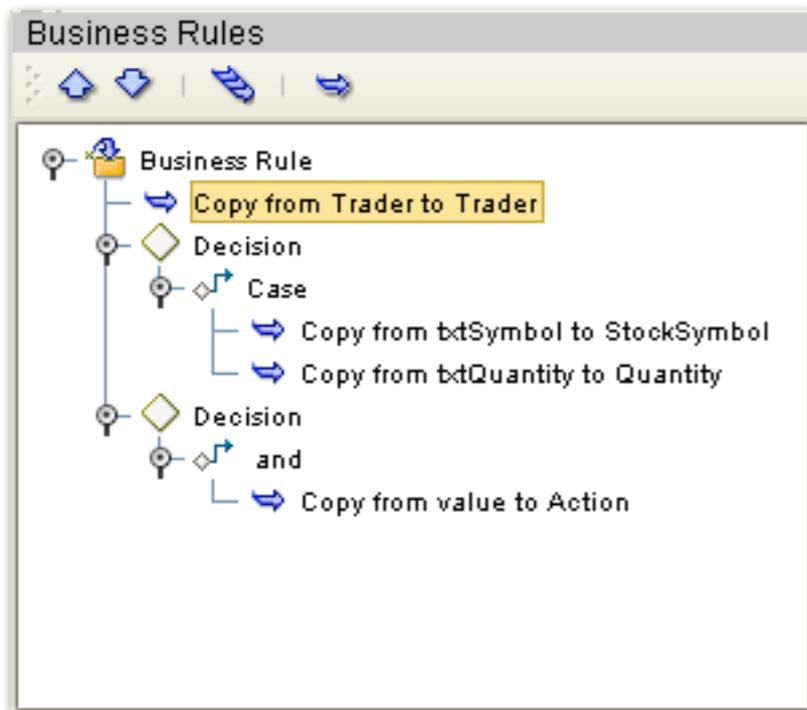
1. There is a LoanApplication [app]
- Any of the following are true:
  - There is an Applicant with:
    - creditRating equal to OK
  - There is an Applicant with:
    - creditRating equal to Sub prime

**THEN**

1. delete LoanApplication [app]
- Set value of LoanApplication [app]
 

approved	false
explanation	Only AA

(show options...)



...NOT ALL GUIs CUT IT...

**Run Buckminster**

Buckminster Installation: buckminster-headless

Target Platform: None

Buckminster Log Level: Info (Default)

Commands:

```

importtargetdefinition -DuseMirror=false -A "${checkout.location}/relog/org.eclipse.xtext.relog/relog/tests/tests-buildserver.target
importprefs -F ${checkout.location}/relog/org.eclipse.xtext.relog/relog/buckminster.workspace.epf

resolve "${checkout.location}/relog/org.eclipse.xtext.relog/relog/tests/xtext-test.mspe"

build

junit --launch "org.eclipse.xtext.common.types.tests/xtext.common.types.tests.launch" --flatXML --output "${WORKSPACE}/test.results/xtext.common.types.tests-${ECLIPSE_TARGET_ALIAS}.xml"
junit --launch "org.eclipse.xtext.purexbase.tests/xtext.purexbase.tests.launch" --flatXML --output "${WORKSPACE}/test.results/xtext.purexbase.tests-${ECLIPSE_TARGET_ALIAS}.xml"
junit --launch "org.eclipse.xtext.xbase.ui.tests/xtext.xbase.ui.tests (xtend).launch" --flatXML --output "${WORKSPACE}/test.results/xtext.xbase.ui.tests.xtend-${ECLIPSE_TARGET_ALIAS}.xml"
junit --launch "org.eclipse.xtext.xtext.ui.tests/xtext.xtext.ui.tests.fast.launch" --flatXML --output "${WORKSPACE}/test.results/xtext.xtext.ui.tests.fast-${ECLIPSE_TARGET_ALIAS}.xml"
junit --launch "org.eclipse.xtext.xtext.ui.graph.tests/xtext.xtext.ui.graph.tests.launch" --flatXML --output

```

Script File:

Advanced...

Delete

Add build step

The screenshot shows the openHAB Designer interface. On the left, there is a 'Configuration' sidebar with a tree view containing 'Items', 'Rules', 'Scripts', and 'Sitemaps'. Below it is an 'Items' sidebar with a tree view containing 'All', 'gC', 'gFF', 'gGF', 'Heating', 'Lights', 'Outdoor', 'Shutters', 'Status', and 'Weather'. The main editor displays a rule named 'demo.rules' with the following code:

```

import org.openhab.core.library.types.*

rule Startup
when
    System started
then
    say("Welcome at openHab!")
end

rule demo
when
    Item Lights received command OFF or
then
end

```

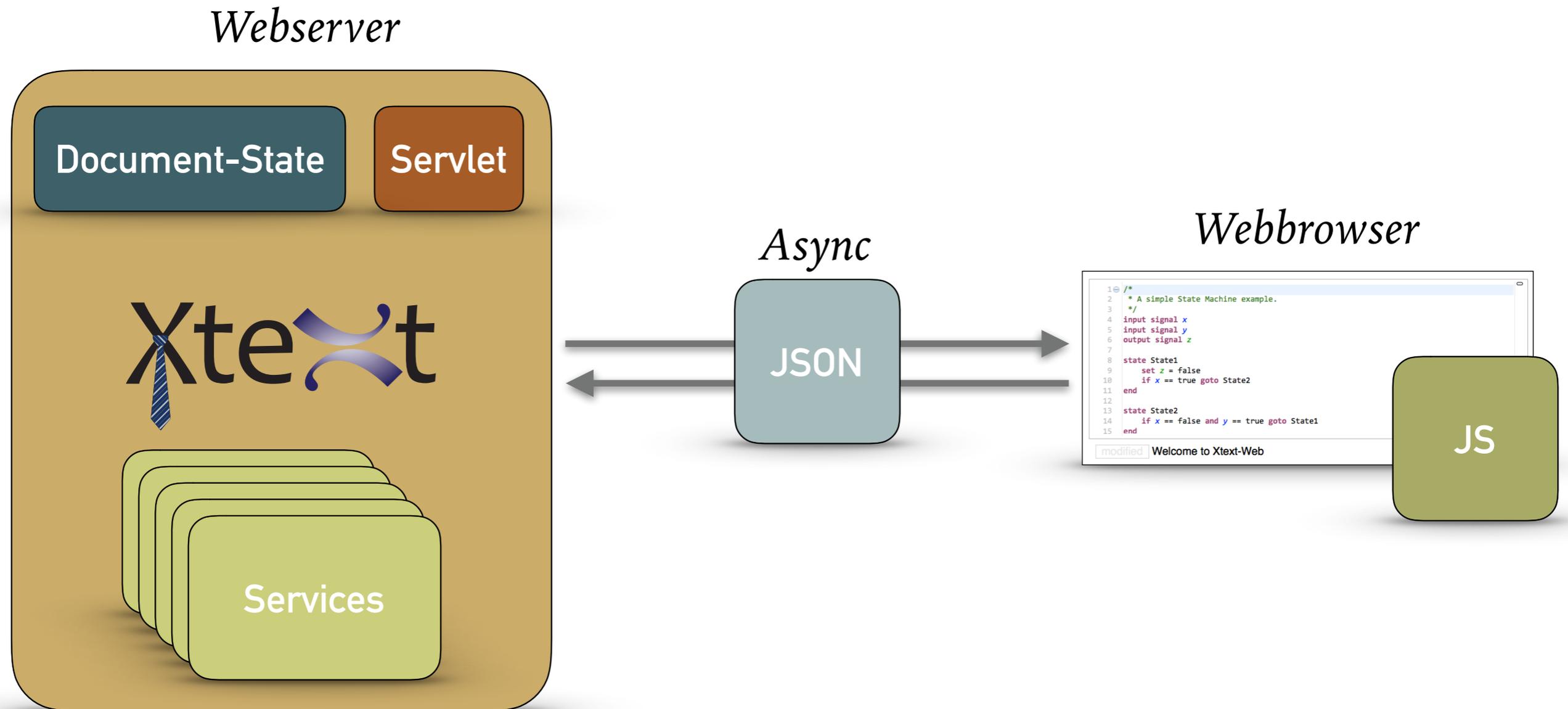
A dropdown menu is open over the 'then' block of the 'demo' rule, showing three options: 'Command Event Trigger - Template for a command tri...', 'State Changed Trigger - Template for state change trig...', and 'State Updated Trigger - Template for state update even...'. The 'Item' option is selected. At the bottom of the dropdown, it says 'Ctrl+Space to show shortest proposals'.

...NOT EVERY CONFIG IS DECLARATIVE

A man with dark hair and a beard, wearing a blue button-down shirt, sits at a desk. He is looking directly at the camera with a bored or unimpressed expression. In front of him is an open newspaper and a glass of dark beer. The background shows a window with a view of a building with a red roof. A white text box with a black border is overlaid on the image, containing the text "TOO LAZY TO DO THE SETUP...".

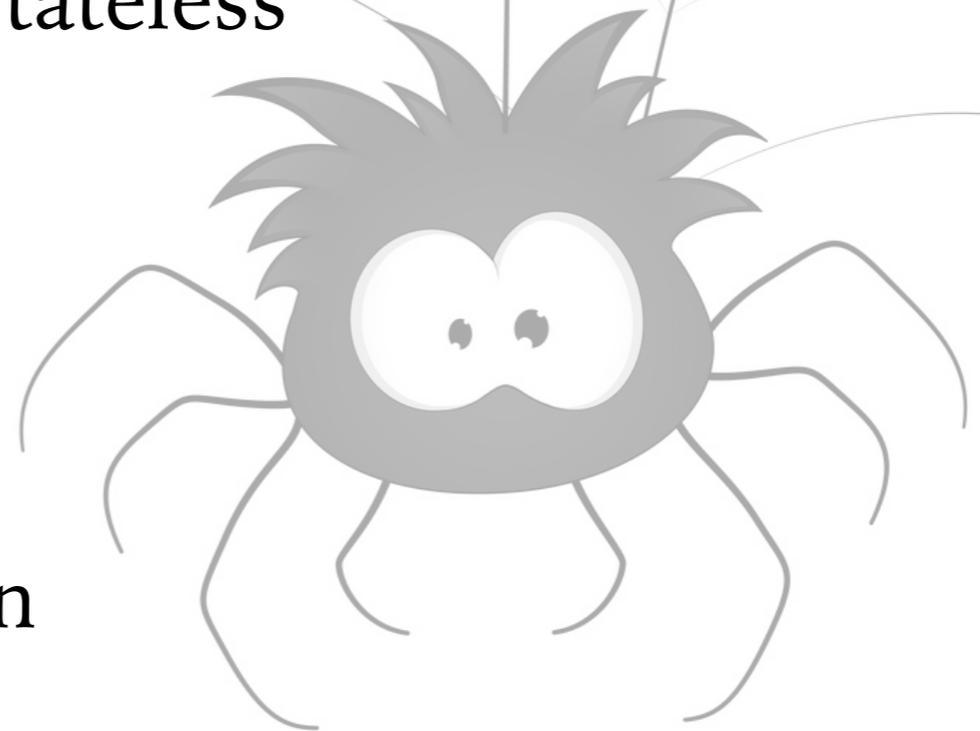
**TOO LAZY TO DO THE SETUP...**

# ARCHITECTURE



# XTEXT IN THE WEB

- Simple servlet-based architecture
  - No Eclipse on server
  - Exchange deltas or complete documents
  - Use file storage on server or stateless
- Single sourcing
  - New project wizard
  - Common parts in \*.ide plug-in
- Gradle or Maven build



# SUPPORTED EDITOR FRONTENDS



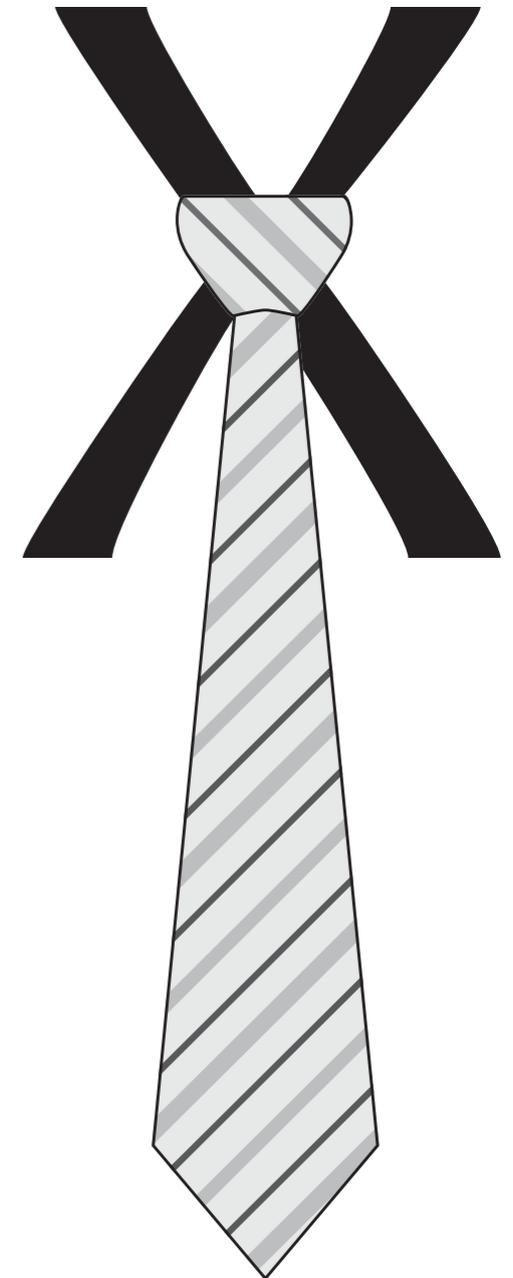
CodeMirror



ORION

# SUPPORTED FEATURES

- Executed on client-side
  - JavaScript based editor widget
  - Syntax highlighting
  - Bracket matching (Orion)
  - Contact server for more elaborate services



# SUPPORTED FEATURES

- Server-side
  - Validation
  - Content assist
  - Semantic highlighting
  - Hover (Orion)
  - Generator
  - Mark occurrences
  - Formatting



**DEMO**