

# IMP: InputLink Manipulation Plug-in

Devvan Stokes

Cory Dunham

# Motivation and Goals

- Input link manipulation is tedious
- “An 80% solution”
  - Specify IL structure or supply VS Datamap
  - Tool generates SML-level code
  - Tweak auto-generated code to suit environment

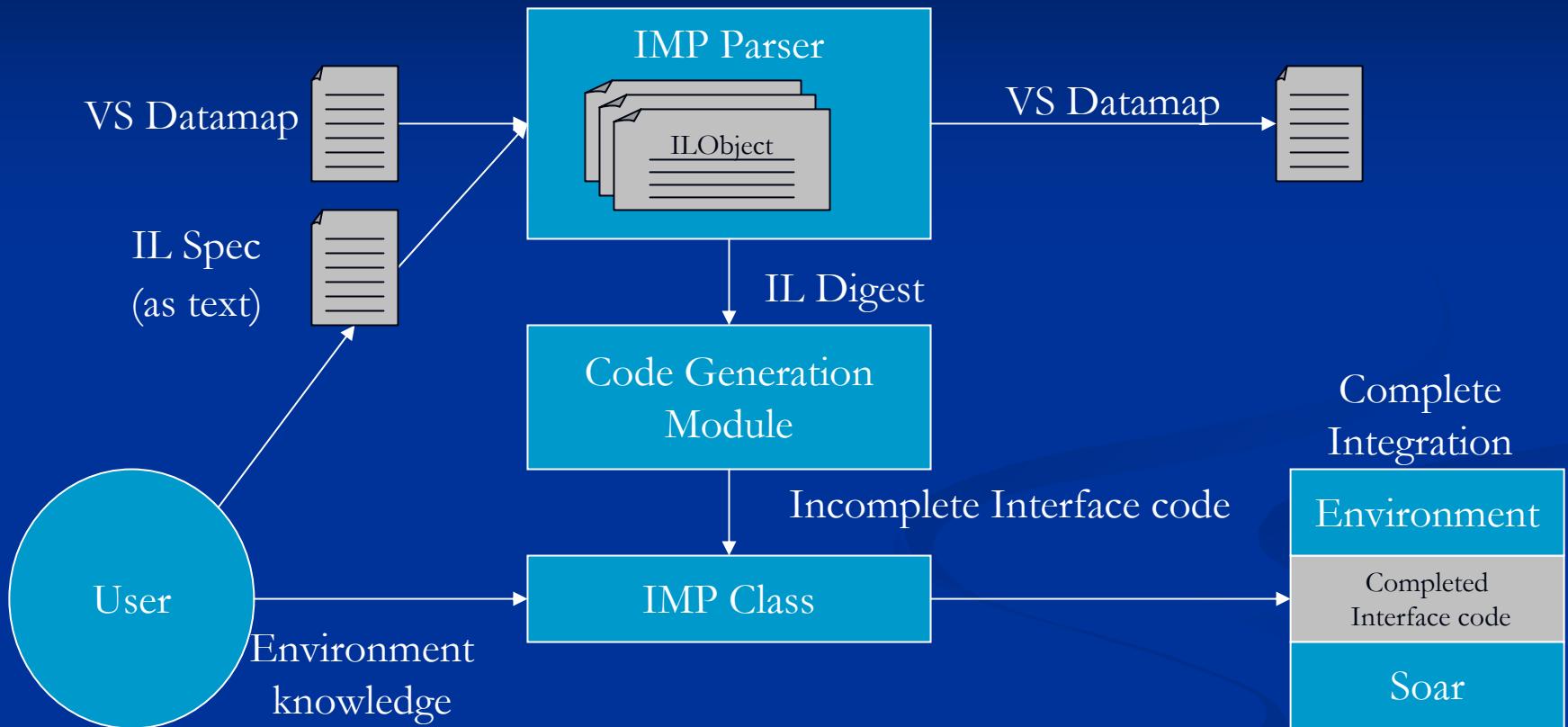
# Interface Code

Environment

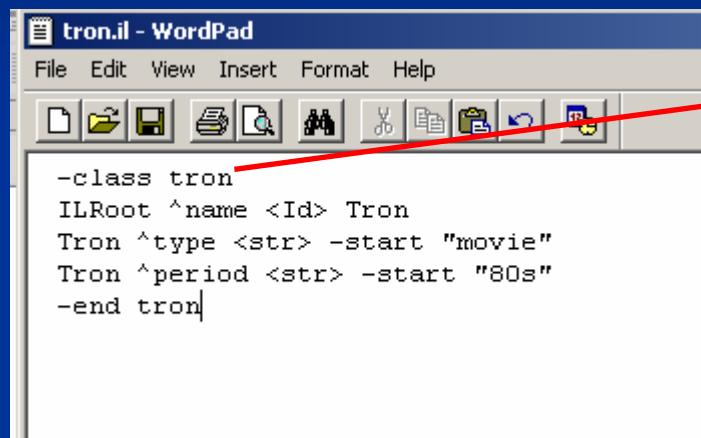
Interface code

Soar

# System Overview

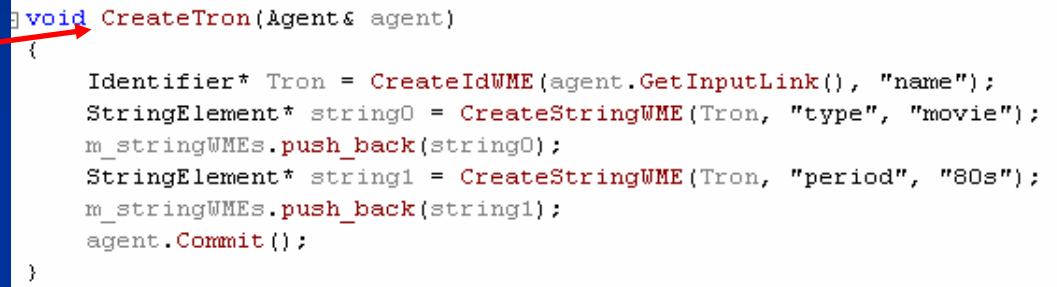


# Example Specification



A screenshot of a Windows WordPad application window titled "tron.il - WordPad". The menu bar includes File, Edit, View, Insert, Format, and Help. The toolbar contains standard file operations like Open, Save, Print, and Find. The main text area contains the following IL script:

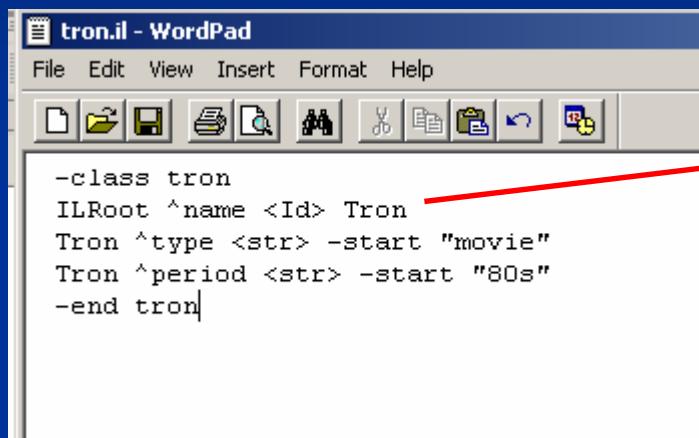
```
-class tron
ILRoot ^name <Id> Tron
Tron ^type <str> -start "movie"
Tron ^period <str> -start "80s"
-end tron
```



A screenshot of a C++ code editor showing a function definition. A red arrow points from the "tron" class definition in the IL script to the first line of the C++ code.

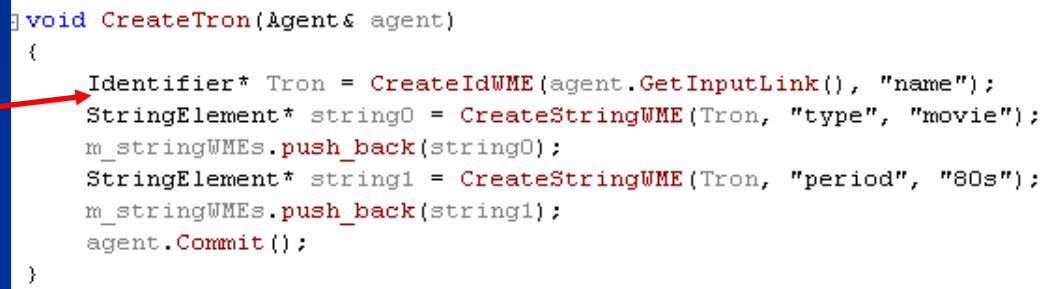
```
void CreateTron(Agent& agent)
{
    Identifier* Tron = CreateIdWME(agent.GetInputLink(), "name");
    StringElement* string0 = CreateStringWME(Tron, "type", "movie");
    m_stringWMEs.push_back(string0);
    StringElement* string1 = CreateStringWME(Tron, "period", "80s");
    m_stringWMEs.push_back(string1);
    agent.Commit();
}
```

# Example Specification



A screenshot of a Windows WordPad application window titled "tron.il - WordPad". The menu bar includes File, Edit, View, Insert, Format, and Help. The toolbar contains standard file operations like Open, Save, Print, and Find. The main text area contains the following IL script:

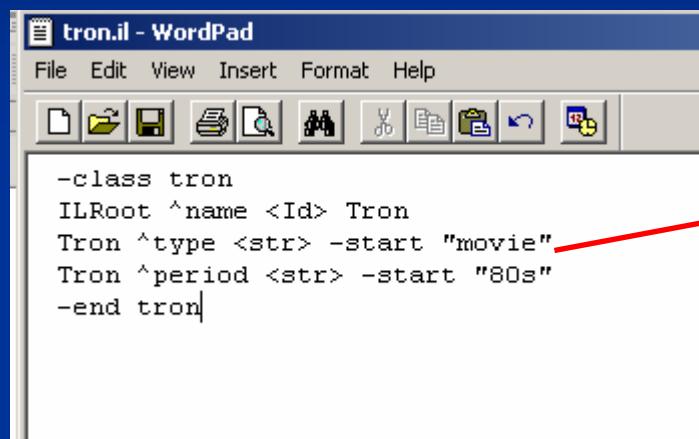
```
-class tron
ILRoot ^name <Id> Tron
Tron ^type <str> -start "movie"
Tron ^period <str> -start "80s"
-end tron
```



A screenshot of a C++ code editor showing the generated C++ code for the Tron class. The code is part of a function named CreateTron. A red arrow points from the word "Tron" in the IL script to the variable declaration "Identifier\* Tron = CreateIdWME(...);".

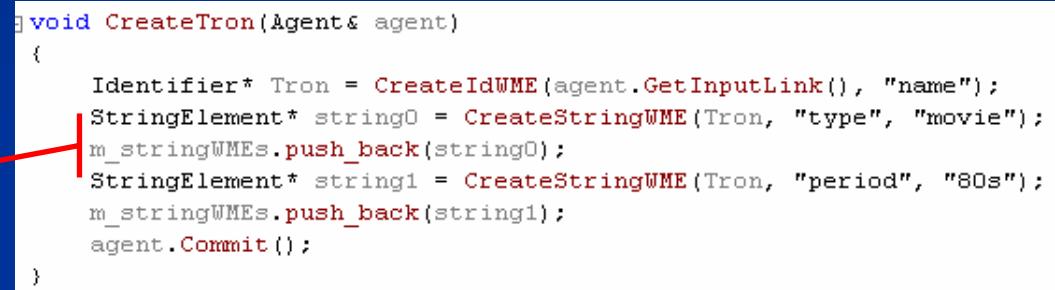
```
void CreateTron(Agent& agent)
{
    Identifier* Tron = CreateIdWME(agent.GetInputLink(), "name");
    StringElement* string0 = CreateStringWME(Tron, "type", "movie");
    m_stringWMEs.push_back(string0);
    StringElement* string1 = CreateStringWME(Tron, "period", "80s");
    m_stringWMEs.push_back(string1);
    agent.Commit();
}
```

# Example Specification



A screenshot of a Windows WordPad application window titled "tron.il - WordPad". The menu bar includes File, Edit, View, Insert, Format, and Help. The toolbar contains standard file operations like Open, Save, Print, and Find. The main text area contains the following IL specification:

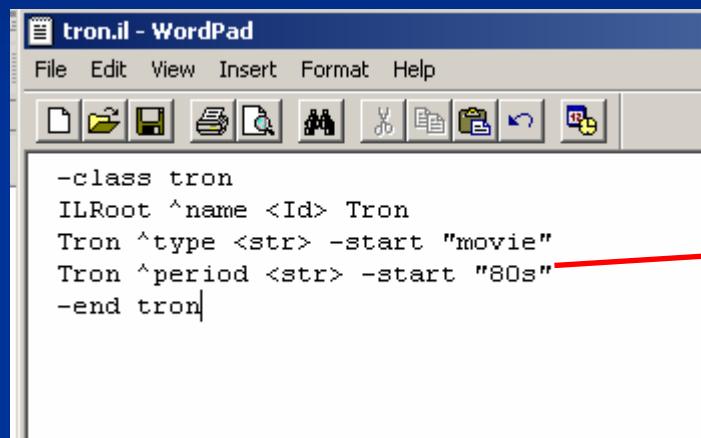
```
-class tron
ILRoot ^name <Id> Tron
Tron ^type <str> -start "movie"
Tron ^period <str> -start "80s"
-end tron
```



A screenshot of a C++ code editor showing generated code for the Tron class. The code is part of a function named CreateTron. A red arrow points from the word "Tron" in the IL specification to the variable "Tron" in the C++ code.

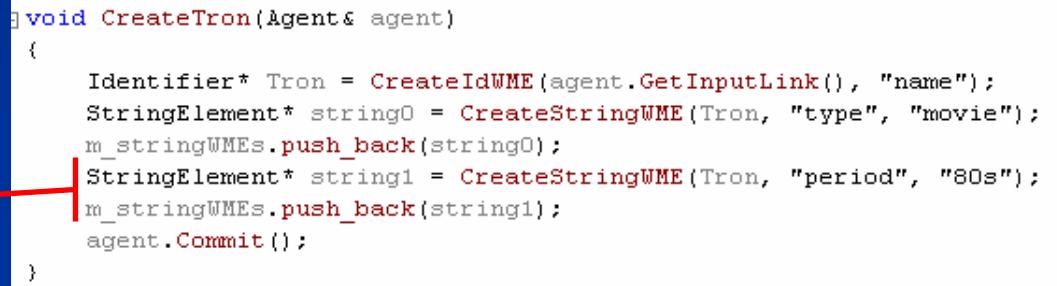
```
void CreateTron(Agent& agent)
{
    Identifier* Tron = CreateIdWME(agent.GetInputLink(), "name");
    StringElement* string0 = CreateStringWME(Tron, "type", "movie");
    m_stringWMEs.push_back(string0);
    StringElement* string1 = CreateStringWME(Tron, "period", "80s");
    m_stringWMEs.push_back(string1);
    agent.Commit();
}
```

# Example Specification



A screenshot of a Windows WordPad application window titled "tron.il - WordPad". The menu bar includes File, Edit, View, Insert, Format, and Help. The toolbar contains standard file operations like Open, Save, Print, and Find. The main text area contains the following IL specification:

```
-class tron
ILRoot ^name <Id> Tron
Tron ^type <str> -start "movie"
Tron ^period <str> -start "80s"
-end tron
```



A screenshot of a C++ code editor showing the generated C++ code for the "tron" class. The code defines a function `CreateTron` that creates an `Identifier` for "name", two `StringElement` objects for "type" and "period", adds them to a vector `m_stringWMEs`, and then commits the changes to an `Agent`.

```
void CreateTron(Agent& agent)
{
    Identifier* Tron = CreateIdWME(agent.GetInputLink(), "name");
    StringElement* string0 = CreateStringWME(Tron, "type", "movie");
    m_stringWMEs.push_back(string0);
    StringElement* string1 = CreateStringWME(Tron, "period", "80s");
    m_stringWMEs.push_back(string1);
    agent.Commit();
}
```

# Current Status

- Can parse the IL specification and generate C++ SML code for IL creation
- Cannot parse datamap, cannot create datamap from spec

# Future Work

- Generate code from VisualSoar datamaps
- Generate Java SML Code
- Create GUI front-end
- Convert to Eclipse Plug-in

# Nuggets & Coal

- Nuggets
  - Higher level of abstraction than SML
- Coal
  - Still in early stages
  - Requires programmer to copy/paste, tweak auto-gen code