libPPF - Programmable Property Files

Table of contents

1. Introduction

Property files are used to associate keys with values. A value can be looked up using the key.

Programmable Property Files are useful when there is a high correlation between the values of the properties that needs to be defined.

PPF uses regular expressions and TCL expressions.

2. PPF Description

Each line in a property file defines one or more properties using regular expressions and TCL expressions.

```
<regexp1> <regexp2> ... <regexpN> = <TCL expression>
```

First, one or more regular expressions defines the property key that must match when the property is looked up. After the '=' sign is a TCL expression that is used to calculate the value of the property.

The variables $1, $2 to $N are bound to the strings that matches each regular expression.

Example:

```
weight 0 = weightless
weight [1-3] = [expr $2 / 2.0]g
```

will represent the properties:

```
weight 0 = weightless
weight 1 = 0.5g
weight 2 = 1g
weight 3 = 1.5g
```

3. libPPF - PPF C++ API

libPPF is an implementation of a PPF API in C++. It consists of three classes, ppf::Parser, ppf::BoundBundle and ppf::Bundle. The latter is not part of the public API.

The API organizes the properties in a tree hierarchy, where BoundBundle represents a node, and which may have properties as leafs.

Example:

```
#include "ppf/parser.h"

using namespace ppf;

Parser parser;
BoundBundle root = parser.parse("properties.ppf");
BoundBundle weight = root.getBoundBundle("weight");
string value = weight.getProperty("2");
```

The parser object must be kept in memory during the time the root bundle and its children are used, because it manages the memory for the parsed data.

4. libPPF Architecture