

# Default skins

## Table of contents

1 Introduction.....	2
2 Convention for choosing skin names.....	2
3 Skin descriptions and examples.....	2
3.1 pelt.....	2
3.2 tigris.....	2
3.3 plain-dev.....	2
4 Old and deprecated skins.....	2
4.1 forrest-site.....	2
4.2 krysalis-site.....	2
4.3 view/viewHelper.....	3

## 1 Introduction

---

Forrest supplies a collection of default skins which are configurable and so should meet the needs of most projects. The aim is to provide many capabilities so that extra skins are not needed.

Note that the new Dispatcher capability in future versions of Forrest will be a better solution.

## 2 Convention for choosing skin names

---

The skin names are based on playing with the word "skin". See our technique for [choosing skin names](#). A name with "-dev" extension signifies that it is under development. There is no concept of versions of default skins. New skins have new names.

## 3 Skin descriptions and examples

---

### 3.1 pelt

---

Uses CSS "div" and no HTML tables.

Examples: [Apache Forrest](#) | [Apache Lenya](#)

### 3.2 tigris

---

This skin is based on version 1.1 of the [style.tigris.org](http://style.tigris.org) project. (It deliberately contravenes our skin naming convention.)

Examples: [Core Computer Security Group](#)

### 3.3 plain-dev

---

This is a very minimal skin to produce plain HTML documents. Such capability might be useful to generate a collection of documents for some off-line product's user help system.

Examples: [snapshot](#)

## 4 Old and deprecated skins

---

The following skins are retained for a little while longer, but are deprecated, so please move to one of the other skins.

### 4.1 forrest-site

---

This is the old skin that we have been dragging around since early days. Uses HTML tables.

Examples: [Apache XML](#)

### 4.2 krysalis-site

---

Uses HTML tables.

Examples:

### 4.3 view/viewHelper

---

This is the evolution of the "leather-dev" skin, to have contracts. It allows the user to provide their own implementations of contracts. The view is controlled by a config file that is easy to understand. It is still in development. Note: you need to have both plugins installed.

Note:

This is now out-of-date. See the new Dispatcher.

Examples: [snapshot](#)