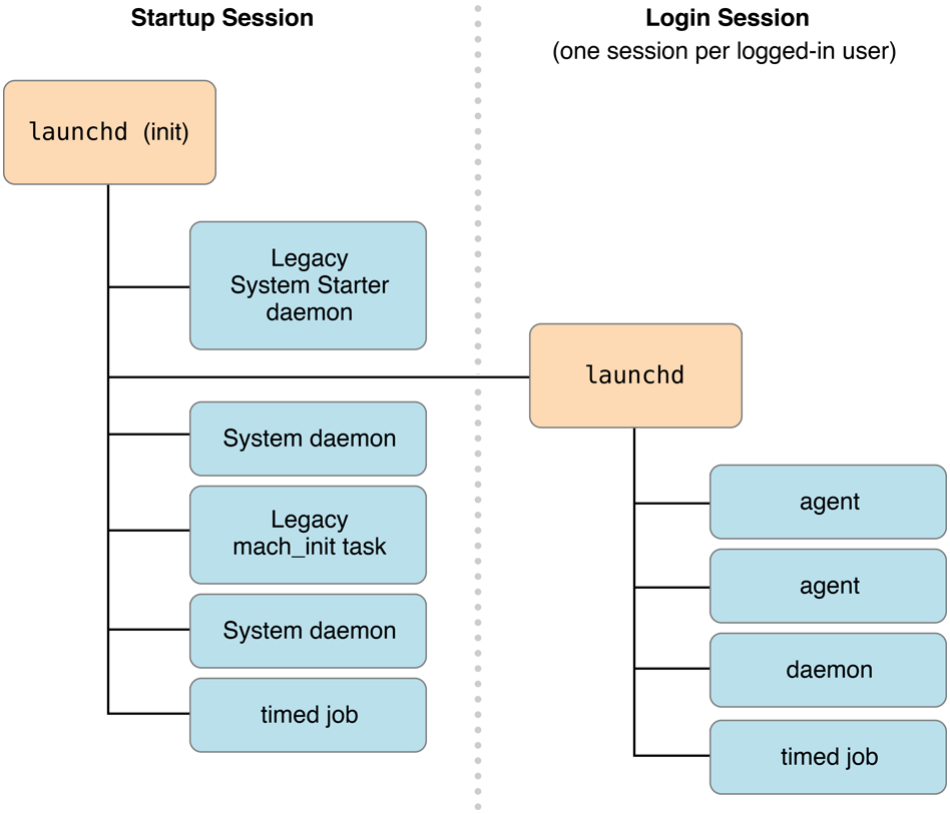
About Daemons and Services

Many kinds of tasks that do not require user interaction are most effectively handled by a process that runs in the background. You can use a daemon or service to:

- Provide server functionality, such serving web pages.
- Coordinate access to of a shared resource, such as a database.
- Perform work for a foreground application, such as file system access.

Figure I-1 Daemons and services are started by launchd in two separate session contexts



Note: This document was previously titled System Startup Programming Topics.

At a Glance

This document provides information that developers of daemons and other low-level system services need to write their code and incorporate it into the startup process. It also provides some useful information for system administrators who must manage the startup process on the computers they manage.

Terminology Note: The terms “service” and “daemon” have several meanings in different contexts, with further variation over time and from one development community to another.

In this document, *service* refers to a background process that supports a full GUI application in some way, for example by registering a global hotkey or by performing network communication. *Daemon* refers to all other types of background processes, especially those that don't present any kind of user interface.

Design your Background Process

OS X provides a variety of background process types with different characteristics, designed for a different situations. There are also several ways for other processes to communicate with background processes. Choosing the appropriate design for a background process is an important first step.

Relevant Chapters: Designing Daemons and Services

Implement your Background Process

Having made the design decisions, you are ready to begin writing code. These chapters guide you through the process of creating specific types of background jobs.

Relevant Chapters: Adding Login Items, Creating XPC Services, Creating Launch Daemons and Agents

Running Jobs on a Timed Schedule

Although it is recommended that background jobs be launched on demand, in some cases running the job on a timed schedule is the most appropriate solution.

Relevant Chapters: Scheduling Timed Jobs

See Also

Logging Reference explains how to use the logging mechanisms provided by the system to assist during debugging and end-user troubleshooting.

Daemons and Agents provides additional details about implementing launch daemons and agents.

Kernel Programming Guide and *Kernel Extension Programming Topics* describe how to write kernel extensions and other kernel-level background processes.

Networking Overview describes the APIs available for sending and receiving data across the network.