

操作系统中进程状态的转换

ZYZ

2018 年 12 月 27 日

1 五种状态的模型

状态 新建 New, 就绪 Ready, 运行 Running, 阻塞 Blocked, 退出 Exit

转换关系

- Admit: 新建的进程被允许, 转换为就绪
- Dispatch: 就绪的进程达到执行条件, 调度为运行
- Timeout: 执行时间超时, 由运行转为就绪
- Event wait: 因等待某个事件的发生或资源的释放, 由运行转为阻塞
- Event occurs: 再次获得资源, 阻塞转换为就绪
- Exit: 正常运行结束, 退出

1.1 DFA自动机

该模型的DFA如图1下所示。

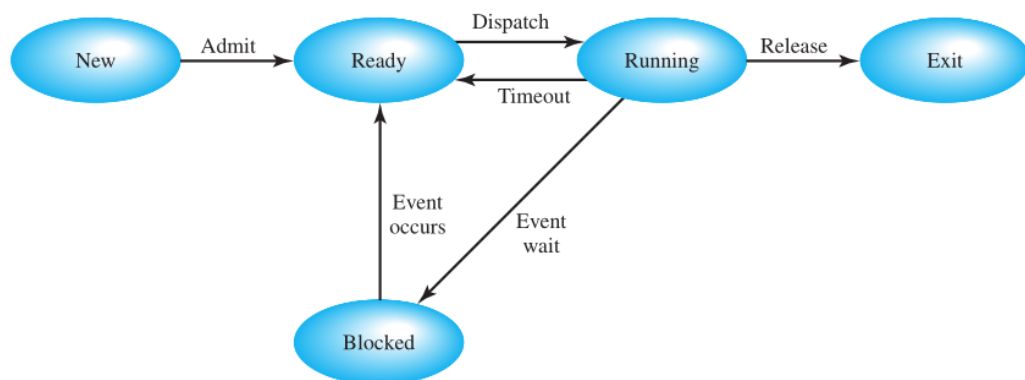


图 1: 五种状态的进程模型DFA

1.2 正则文法

粗体代表非终结符，普通的为终结符，**New**为开始符号

New \rightarrow *admit***Ready**
Ready \rightarrow *dispatch***Running**
Running \rightarrow *timeout***Ready**
Running \rightarrow *event_wait***Blocked**
Blocked \rightarrow *event_occurs***Ready**
Running \rightarrow *release***Exit**
Exit \rightarrow *exit*

2 含有一个挂起状态

状态 新建 **New**，就绪 **Ready**，运行 **Running**，阻塞 **Blocked**，挂起 **Suspend**，退出 **Exit**

转换关系

- **Admit**: 新建的进程被允许，转换为就绪
- **Dispatch**: 就绪的进程达到执行条件，调度为运行
- **Timeout**: 执行时间超时，由运行转为就绪

- Event wait: 因等待某个事件的发生或资源的释放, 由运行转为阻塞
- Event occurs: 再次获得资源, 阻塞转换为就绪
- Suspend: 暂停阻塞的进程, 静止下来, 转换为挂起
- Activate: 激活挂起的进程, 转换为就绪
- Exit: 正常运行结束, 退出

2.1 DFA自动机

该模型的DFA如图2所示。

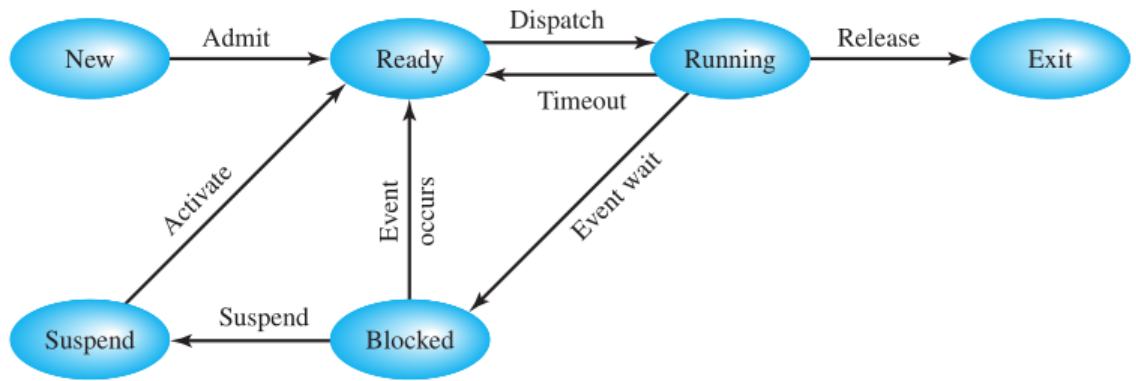


图 2: 六种状态的进程模型DFA

2.2 正则文法

粗体代表非终结符, 普通的为终结符, **New**为开始符号

New \rightarrow *admit***Ready**
Ready \rightarrow *dispatch***Running**
Running \rightarrow *timeout***Ready**
Running \rightarrow *event_wait***Blocked**
Blocked \rightarrow *event_occurs***Ready**
Blocked \rightarrow *suspend***Suspend**
Suspend \rightarrow *activate***Ready**
Running \rightarrow *release***Exit**
Exit \rightarrow *exit*

3 含有两个挂起状态

状态 新建 New, 就绪 Ready, 就绪挂起 Ready Suspend, 运行 Running, 阻塞 Blocked, 阻塞挂起 Blocked Suspend, 退出 Exit

转换关系

- Admit: 新建的进程被允许, 转换为就绪
- Dispatch: 就绪的进程达到执行条件, 调度为运行
- Suspend: 暂停进程的运行, 挂起以释放内存, 由就绪、运行或者阻塞转换为相应的挂起
- Activate: 激活挂起的进程, 由就绪挂起或者阻塞挂起转换为就绪
- Timeout: 执行时间超时, 由运行转为就绪
- Event wait: 因等待某个事件的发生或资源的释放, 由运行转为阻塞
- Event occurs: 再次获得资源, 阻塞转换为就绪, 或者阻塞挂起转换为就绪挂起
- Exit: 正常运行结束, 退出

3.1 NFA自动机

该模型的NFA如图3所示。

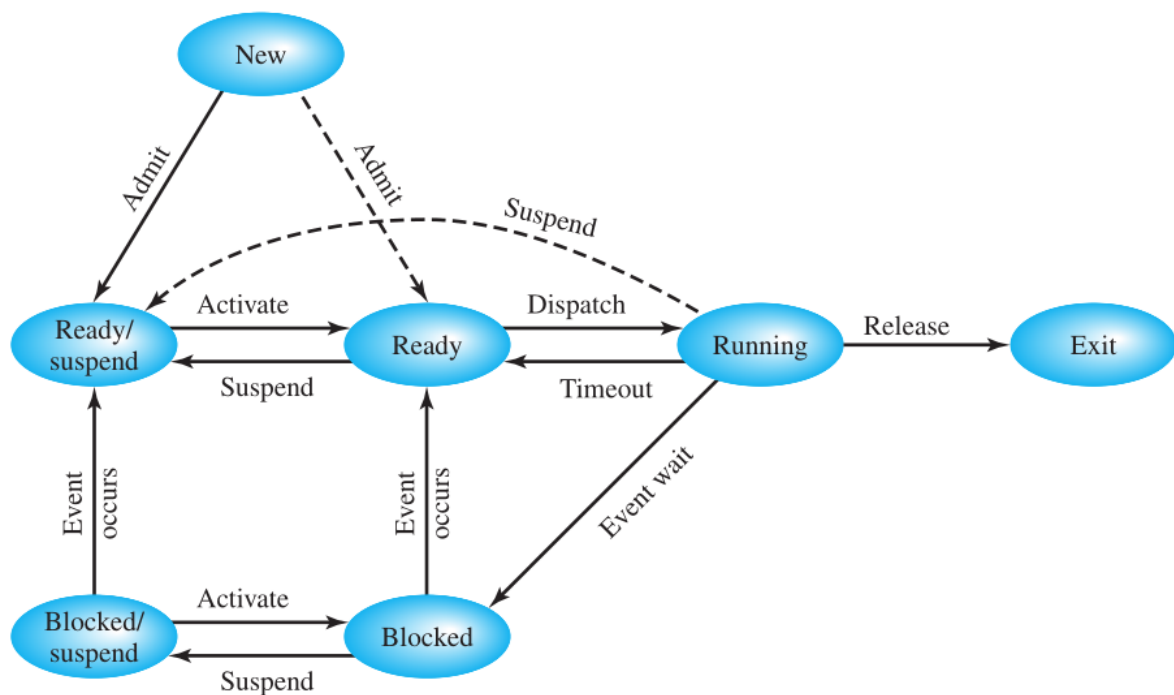


图 3: 七种状态的进程模型NFA

3.2 正则文法

粗体代表非终结符，普通的为终结符，**New**为开始符号

New \rightarrow *admit***Ready**
New \rightarrow *admit***Ready_suspend**
Ready_suspend \rightarrow *r_activate***Ready**
Ready \rightarrow *re_suspend***Ready_suspend**
Ready \rightarrow *dispatch***Running**
Running \rightarrow *timeout***Ready**
Running \rightarrow *event_wait***Blocked**
Running \rightarrow *ru_suspend***Ready_suspend**
Blocked \rightarrow *b_event_occurs***Ready**
Blocked \rightarrow *b_suspend***Blocked_suspend**
Blocked_suspend \rightarrow *b_activate***Blocked**
Blocked_suspend \rightarrow *bs_event_occurs***Ready_suspend**

Running \rightarrow *release***Exit**
Exit \rightarrow *exit*

参考文献

- [1] tallings W , Education P . Operating Systems: Internals and Design Principles:International Edition[J]. Bbc Video Library, 2008, 18(2):235–244.