

14 过程锁

过程锁类似于数据库的独占锁机制，谁加的锁必须由谁来解锁，除非是：

- 超级管理员（`Phenix.Business.Security.UserPrincipal.User.IsAdminRole = true`）；
- 上级（解锁者在加锁者的 `Phenix.Business.Security.UserPrincipal.User.AllSuperiors` 中）；
- 锁已超期（`ProcessLockInfo.ExpiryTime < DateTime.Now`）；

对于无有效期的锁（加锁时 `expiryTime` 参数为 `TimeSpan.Zero`），必须由自己或上级、超级管理员才能解锁。

过程锁的应用，可以大到一个功能模块的锁定，小到一个业务对象、属性、过程的锁定。

我们可以把过程锁想象为一种跨物理域的 lock 语句，对于无有效期的锁，加锁和解锁的代码应该一一对应，且能保证执行到，特别是解锁代码。

由于过程锁提供了无限可能的应用方式，需要结合实际业务需求有针对性地设计，合理使用。

14.1 案例

举一个案例来说明过程锁的应用——“独占窗体”。

这是一个假设性的应用场景，仅允许唯一一个操作者操作窗体，在他未关闭窗体前，其他操作者是无法打开它的，而且系统会提示信息后自动关闭它：

```
private void ProcessLockManageForm_Shown(object sender, EventArgs e)
{
    try
    {
        Phenix.Business.Security.UserPrincipal.User.Identity.LockProcess(this.GetType().FullName,
this.Text, "本窗体解锁后才能打开!");
    }
    catch (Phenix.Core.Security.ProcessLockException ex)
    {
        MessageBox.Show(ex.Message, this.Text, MessageBoxButtons.OK, MessageBoxIcon.Warning);
        this.Close();
    }
}

private void ProcessLockManageForm_FormClosed(object sender, FormClosedEventArgs e)
{
    try
    {
        Phenix.Business.Security.UserPrincipal.User.Identity.UnlockProcess(this.GetType().FullName);
    }
    catch (Phenix.Core.Security.ProcessLockException)
```

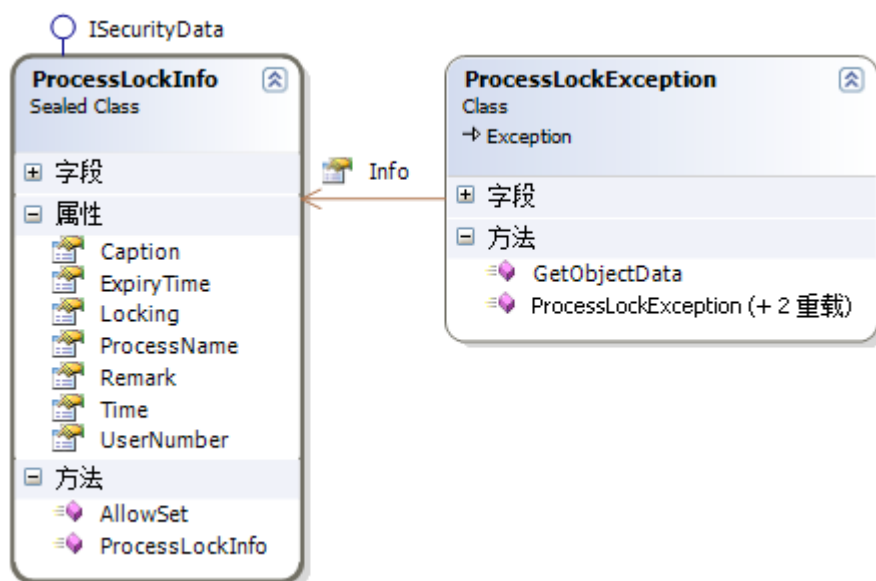
```
{  
}  
}
```

在窗体锁定阶段，其他操作者要尝试打开它时，都会弹出如下提示：



14.2 过程锁信息和过程锁异常

上述案例中，提示信息显示的是抛出异常（`Phenix.Core.Security.ProcessLockException`）中的 `Message` 属性值，这是根据其 `Info` 属性值（`ProcessLockInfo` 对象）拼接出来的字符串，具体应用可以直接通过这 `Info` 属性内容，自行拼接出更加友好的提示信息。



14.3 加、解锁

`Phenix.Business.Security.UserPrincipal.User` 代表了当前登录的用户，其 `Identity` 属性是 `Phenix.Core.Security.UserIdentity` 类的实例，通过它提供的如下函数，我们可实现过程锁的加、解锁：

```
/// <summary>
/// 过程加锁
/// </summary>
/// <param name="processName">过程名</param>
/// <param name="caption">标签</param>
/// <param name="expiryTime">限期, TimeSpan.Zero代表不自动失效</param>
/// <param name="remark">备注</param>
public void LockProcess(string processName, string caption, TimeSpan expiryTime, string remark)

/// <summary>
/// 过程解锁
/// </summary>
/// <param name="processName">过程名</param>
/// <param name="remark">备注</param>
public void UnlockProcess(string processName, string remark)
```

14.4 取过程锁信息

```
/// <summary>
/// 取过程锁资料
/// </summary>
/// <param name="processName">过程名</param>
public ProcessLockInfo GetProcessLockInfo(string processName)
```

14.5 判断是否允许操作过程锁

```
/// <summary>
/// 是否允许操作过程锁
/// </summary>
/// <param name="processName">过程名</param>
public bool AllowSetProcessLockInfo(string processName)

/// <summary>
/// 是否允许操作过程锁
/// </summary>
/// <param name="info">过程锁资料</param>
public bool AllowSetProcessLockInfo(ProcessLockInfo info)
```