LW\_NTSTATUS

LwRtlFiberCreate(

LW\_IN LW\_FIBER\_FUNCTION Start,

LW\_IN LW\_PVOID pParam,

LW\_IN LW\_ULONG StackSize,

LW\_IN LW\_OPTIONAL PLW\_FIBER\_SCHEDULER Scheduler,

LW\_IN LW\_OPTIONAL LW\_PVOID SchedulerData,

LW\_OUT PLW\_FIBER\* ppFiber

)

{

pFiber->RefCount = 1;

pFiber->Id = LwInterlockedIncrement(&NextFiberId);

#ifdef HAVE\_VALGRIND\_VALGRIND\_H

// Tell valgrind about our stack so it doesn't get

// confused on stack pointer changes in ContextSwitch

pFiber->ValgrindId = VALGRIND\_STACK\_REGISTER(

pFiber->pStackBase,

pFiber->pStackBase + StackSize);

#endif

// Prepare context for start function to run

PrepareContext(

&pFiber->MachineContext,

pFiber->pStackBase,

StackSize,

Start,

pParam);

// Return to the current fiber by default

LwRtlFiberSetReturn(pFiber, GetCurrentFiber());

// Insert fiber into global list

LwRtlSpinlockLock(&spinLock);

LwListInsertTail(&allFibers, &pFiber->AllEntry);

LwRtlSpinlockUnlock(&spinLock);

cleanup:

if (status && pFiber)

{

LwRtlFiberRelease(pFiber);

pFiber = NULL;

}

\*ppFiber = pFiber;

return status;

}