

I've got it! That's what will work:

```
struct task_struct {
    ...
    struct beancounter *exec_bc;
    struct beancounter *tmp_exec_bc; /* is set to NULL on
                                     * tsk creation
                                     */
};

struct beancounter get_exec_bc(void)
{
    if (current->tmp_exec_bc)
        return current->tmp_exec_bc;
    return rcu_dereference(current->exec_bc);
}

struct beancounter set_tmp_exec_bc(struct beancounter *new)
{
    struct beancounter *old;

    old = current->tmp_exec_bc;
    current->tmp_exec_bc = new;
    return old;
}

void reset_tmp_exec_bc(struct beancounter *expected_old)
{
    BUG_ON(current->tmp_exec_bc != expected_old);
    current->tmp_exec_bc = NULL;
}

void move_task(struct task_struct *tsk, struct beancounter *bc)
{
    struct beancounter *old;

    mutex_lock(&tsk_move_mutex);
    old = tsk->exec_bc;
    get_bc(bc);
    rcu_assign_pointer(current->exec_bc, bc);
    synchronize_rcu();
    mutex_unlock(&tsk_move_mutex);
}
```

```
bc_put(old);  
}
```

I will implement this in the next beancounter patches.
Thanks for discussion :)
