Multiple Rectangle Cropping

RFC to extend the selection API
Current Capabilities

DATA SOURCE

CROP_DEFAULT
CROP_ACTIVE
CROP_BOUNDS

DATA SINK

COMPOSE_DEFAULT
COMPOSE_ACTIVE
COMPOSE_BOUNDS
COMPOSE_PADDED

overscan area
Proposed Capabilities

DATA SOURCE

CROP_DEFAULT

DATA SINK

COMPOSE_DEFAULT

overscan area

CROP_BOUNDS

COMPOSE_BOUNDS
Applications
Support in sensors

**Figure 39:** Example of 4 multiple frames read-out
Why not in userland?

- Increase FPS
- Reduce data transfer

- More potatoes per minute
- More accurate speed tickets :P
- Reduce power consumption
struct v4l2_subdev_selection {
    __u32 pad;
    __u32 target;
    __u32 flags;
-    struct v4l2_rect r;
-    __u32 reserved[8];
+    union {
+        struct v4l2_rect r;
+        struct v4l2_ext_rect *pr;
+    };
    __u32 rectangles;
    __u32 reserved[7];
};

struct v4l2_ext_rect {
    __s32 left;
    __s32 top;
    __u32 width;
    __u32 height;
    __u32 reserved[4];
};

+ Helpers: multi to legacy, legacy to multi
+ ioctl32
Why v4l2_ext_rectangle?

- Opportunity to remove sign on sizes.
- Opportunity to add fields to the rectangles (ie tracking areas)