

Wayland IVI Extension Design IVI input

[Getting Started with new Input Handling APIs](#)

Input handling interfaces

No	Name	Definition	Parameters	Description
1	ilm_setInputAcceptanceOn	ilmErrorTypes	t_ilm_surface surfaceID, t_ilm_uint num_seats, t_ilm_string *seats	<p>\brief Set the surface's accepted seats to the list specified</p> <pre> \ingroup ilmControl \param[in] surfaceID The surface whose list of accepted seats is to be changed \param[in] num_seats The number of seats stored in seats \param[in] seats A pointer to an array of strings listing each seat to accept by its seat name \return ILM_SUCCESS if the method call was successful \return ILM_FAILED if the client cannot call the method on the surface </pre>
2	ilm_getInputAcceptanceOn	ilmErrorTypes	t_ilm_surface surfaceID, t_ilm_uint *num_seats, t_ilm_string **seats	<p>\brief Get the list of seats that support the device types specified in bitmask</p> <pre> \ingroup ilmControl \param[in] surfaceID The surface that the list of seats comes from \param[out] num_seats The number of seats returned \param[out] seats A pointer to the memory where an array of seats is stored. It is the caller's responsibility to free this memory after use. \return ILM_SUCCESS if the method call was successful \return ILM_FAILED if the client cannot call the method on the surface </pre>
3	ilm_getInputDevices	ilmErrorTypes	ilmInputDevice bitmask, t_ilm_uint *num_seats, t_ilm_string **seats	<p>\brief Get the list of seats that support the device types specified in bitmask</p> <pre> \ingroup ilmControl \param[in] bitmask The bitmask that indicates what type of devices are requested (e.g. ILM_INPUT_DEVICE_POINTER) \param[out] num_seats The number of seats returned \param[out] seats A pointer to the memory where an array of seats is stored. It is the caller's responsibility to free this memory after use. \return ILM_SUCCESS if the method call was successful \return ILM_FAILED if the method call was unsuccessful </pre>
4	ilm_getInputDeviceCapabilities	ilmErrorTypes	t_ilm_string seat_name, ilmInputDevice* bitmask	<p>\brief Set whether the specified surfaces have input focus set for the given device types</p> <pre> \ingroup ilmControl \param[in] seat_name The name of the seat \param[out] bitmask A pointer to the bitmask that should be set \return ILM_SUCCESS if the method call was successful \return ILM_FAILED if the method call was unsuccessful </pre>
5	ilm_setInputFocus	ilmErrorTypes	t_ilm_surface *surfaceIDs, t_ilm_uint num_surfaces, ilmInputDevice bitmask, t_ilm_bool is_set	<p>\brief Set whether the specified surfaces have input focus set for the given device types</p> <pre> \ingroup ilmControl \param[in] surfaceIDs An array of surface IDs whose input focus may be changed \param[in] num_surfaces The number of surfaces in surfaceIDs \param[in] bitmask A bitmask of the types of device for which focus will be set \param[in] is_set ILM_TRUE if focus is to be set, ILM_FALSE if focus is to be unset \return ILM_SUCCESS if the method call was successful \return ILM_FAILED if the method call was unsuccessful </pre>

6	ilm_getInput Focus	ilmErrorTypes	t_ilm_surface **surfaceIDs, ilmInputDevice** bitmasks, t_ilm_uint *num_ids	<p>\brief Get all surface IDs and their corresponding focus bitmasks</p> <pre> \ingroup ilmControl \param[out] surfaceIDs A pointer to the memory where an array of surface IDs will be created. The caller is responsible for freeing this memory after use. \param[out] bitmasks A pointer to the memory where an array of bitmasks will be created. The caller is responsible for freeing this memory after use. \param[out] num_ids The number of surface IDs that were returned \return ILM_SUCCESS if the method call was successful \return ILM_FAILED if the method call was unsuccessful </pre>
---	-----------------------	---------------	--	--

Copied from MediaWiki
Last Edit: 15:30, 30 July 2015 Emre.ucan