;; Auto-generated. Do not edit!

(when (boundp 'gazebo\_msgs\_new::SetModelConfiguration)

(if (not (find-package "GAZEBO\_MSGS\_NEW"))

(make-package "GAZEBO\_MSGS\_NEW"))

(shadow 'SetModelConfiguration (find-package "GAZEBO\_MSGS\_NEW")))

(unless (find-package "GAZEBO\_MSGS\_NEW::SETMODELCONFIGURATION")

(make-package "GAZEBO\_MSGS\_NEW::SETMODELCONFIGURATION"))

(unless (find-package "GAZEBO\_MSGS\_NEW::SETMODELCONFIGURATIONREQUEST")

(make-package "GAZEBO\_MSGS\_NEW::SETMODELCONFIGURATIONREQUEST"))

(unless (find-package "GAZEBO\_MSGS\_NEW::SETMODELCONFIGURATIONRESPONSE")

(make-package "GAZEBO\_MSGS\_NEW::SETMODELCONFIGURATIONRESPONSE"))

(in-package "ROS")

(defclass gazebo\_msgs\_new::SetModelConfigurationRequest

:super ros::object

:slots (\_model\_name \_urdf\_param\_name \_joint\_names \_joint\_positions ))

(defmethod gazebo\_msgs\_new::SetModelConfigurationRequest

(:init

(&key

((:model\_name \_\_model\_name) "")

((:urdf\_param\_name \_\_urdf\_param\_name) "")

((:joint\_names \_\_joint\_names) (let (r) (dotimes (i 0) (push "" r)) r))

((:joint\_positions \_\_joint\_positions) (make-array 0 :initial-element 0.0 :element-type :float))

)

(send-super :init)

(setq \_model\_name (string \_\_model\_name))

(setq \_urdf\_param\_name (string \_\_urdf\_param\_name))

(setq \_joint\_names \_\_joint\_names)

(setq \_joint\_positions \_\_joint\_positions)

self)

(:model\_name

(&optional \_\_model\_name)

(if \_\_model\_name (setq \_model\_name \_\_model\_name)) \_model\_name)

(:urdf\_param\_name

(&optional \_\_urdf\_param\_name)

(if \_\_urdf\_param\_name (setq \_urdf\_param\_name \_\_urdf\_param\_name)) \_urdf\_param\_name)

(:joint\_names

(&optional \_\_joint\_names)

(if \_\_joint\_names (setq \_joint\_names \_\_joint\_names)) \_joint\_names)

(:joint\_positions

(&optional \_\_joint\_positions)

(if \_\_joint\_positions (setq \_joint\_positions \_\_joint\_positions)) \_joint\_positions)

(:serialization-length

()

(+

;; string \_model\_name

4 (length \_model\_name)

;; string \_urdf\_param\_name

4 (length \_urdf\_param\_name)

;; string[] \_joint\_names

(apply #'+ (mapcar #'(lambda (x) (+ 4 (length x))) \_joint\_names)) 4

;; float64[] \_joint\_positions

(\* 8 (length \_joint\_positions)) 4

))

(:serialize

(&optional strm)

(let ((s (if strm strm

(make-string-output-stream (send self :serialization-length)))))

;; string \_model\_name

(write-long (length \_model\_name) s) (princ \_model\_name s)

;; string \_urdf\_param\_name

(write-long (length \_urdf\_param\_name) s) (princ \_urdf\_param\_name s)

;; string[] \_joint\_names

(write-long (length \_joint\_names) s)

(dolist (elem \_joint\_names)

(write-long (length elem) s) (princ elem s)

)

;; float64[] \_joint\_positions

(write-long (length \_joint\_positions) s)

(dotimes (i (length \_joint\_positions))

(sys::poke (elt \_joint\_positions i) (send s :buffer) (send s :count) :double) (incf (stream-count s) 8)

)

;;

(if (null strm) (get-output-stream-string s))))

(:deserialize

(buf &optional (ptr- 0))

;; string \_model\_name

(let (n) (setq n (sys::peek buf ptr- :integer)) (incf ptr- 4) (setq \_model\_name (subseq buf ptr- (+ ptr- n))) (incf ptr- n))

;; string \_urdf\_param\_name

(let (n) (setq n (sys::peek buf ptr- :integer)) (incf ptr- 4) (setq \_urdf\_param\_name (subseq buf ptr- (+ ptr- n))) (incf ptr- n))

;; string[] \_joint\_names

(let (n)

(setq n (sys::peek buf ptr- :integer)) (incf ptr- 4)

(setq \_joint\_names (make-list n))

(dotimes (i n)

(let (n) (setq n (sys::peek buf ptr- :integer)) (incf ptr- 4) (setf (elt \_joint\_names i) (subseq buf ptr- (+ ptr- n))) (incf ptr- n))

))

;; float64[] \_joint\_positions

(let (n)

(setq n (sys::peek buf ptr- :integer)) (incf ptr- 4)

(setq \_joint\_positions (instantiate float-vector n))

(dotimes (i n)

(setf (elt \_joint\_positions i) (sys::peek buf ptr- :double)) (incf ptr- 8)

))

;;

self)

)

(defclass gazebo\_msgs\_new::SetModelConfigurationResponse

:super ros::object

:slots (\_success \_status\_message ))

(defmethod gazebo\_msgs\_new::SetModelConfigurationResponse

(:init

(&key

((:success \_\_success) nil)

((:status\_message \_\_status\_message) "")

)

(send-super :init)

(setq \_success \_\_success)

(setq \_status\_message (string \_\_status\_message))

self)

(:success

(&optional \_\_success)

(if \_\_success (setq \_success \_\_success)) \_success)

(:status\_message

(&optional \_\_status\_message)

(if \_\_status\_message (setq \_status\_message \_\_status\_message)) \_status\_message)

(:serialization-length

()

(+

;; bool \_success

1

;; string \_status\_message

4 (length \_status\_message)

))

(:serialize

(&optional strm)

(let ((s (if strm strm

(make-string-output-stream (send self :serialization-length)))))

;; bool \_success

(if \_success (write-byte -1 s) (write-byte 0 s))

;; string \_status\_message

(write-long (length \_status\_message) s) (princ \_status\_message s)

;;

(if (null strm) (get-output-stream-string s))))

(:deserialize

(buf &optional (ptr- 0))

;; bool \_success

(setq \_success (not (= 0 (sys::peek buf ptr- :char)))) (incf ptr- 1)

;; string \_status\_message

(let (n) (setq n (sys::peek buf ptr- :integer)) (incf ptr- 4) (setq \_status\_message (subseq buf ptr- (+ ptr- n))) (incf ptr- n))

;;

self)

)

(defclass gazebo\_msgs\_new::SetModelConfiguration

:super ros::object

:slots ())

(setf (get gazebo\_msgs\_new::SetModelConfiguration :md5sum-) "10e3139d3b669c40afc057d38956fff7")

(setf (get gazebo\_msgs\_new::SetModelConfiguration :datatype-) "gazebo\_msgs\_new/SetModelConfiguration")

(setf (get gazebo\_msgs\_new::SetModelConfiguration :request) gazebo\_msgs\_new::SetModelConfigurationRequest)

(setf (get gazebo\_msgs\_new::SetModelConfiguration :response) gazebo\_msgs\_new::SetModelConfigurationResponse)

(defmethod gazebo\_msgs\_new::SetModelConfigurationRequest

(:response () (instance gazebo\_msgs\_new::SetModelConfigurationResponse :init)))

(setf (get gazebo\_msgs\_new::SetModelConfigurationRequest :md5sum-) "10e3139d3b669c40afc057d38956fff7")

(setf (get gazebo\_msgs\_new::SetModelConfigurationRequest :datatype-) "gazebo\_msgs\_new/SetModelConfigurationRequest")

(setf (get gazebo\_msgs\_new::SetModelConfigurationRequest :definition-)

"# Set Gazebo Model pose and twist

string model\_name # model to set state (pose and twist)

string urdf\_param\_name # parameter name that contains the urdf XML.

string[] joint\_names # list of joints to set positions. if joint is not listed here, preserve current position.

float64[] joint\_positions # set to this position.

---

bool success # return true if setting state successful

string status\_message # comments if available

")

(setf (get gazebo\_msgs\_new::SetModelConfigurationResponse :md5sum-) "10e3139d3b669c40afc057d38956fff7")

(setf (get gazebo\_msgs\_new::SetModelConfigurationResponse :datatype-) "gazebo\_msgs\_new/SetModelConfigurationResponse")

(setf (get gazebo\_msgs\_new::SetModelConfigurationResponse :definition-)

"# Set Gazebo Model pose and twist

string model\_name # model to set state (pose and twist)

string urdf\_param\_name # parameter name that contains the urdf XML.

string[] joint\_names # list of joints to set positions. if joint is not listed here, preserve current position.

float64[] joint\_positions # set to this position.

---

bool success # return true if setting state successful

string status\_message # comments if available

")

(provide :gazebo\_msgs\_new/SetModelConfiguration "10e3139d3b669c40afc057d38956fff7")