;; Auto-generated. Do not edit!

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

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

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

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

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

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

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

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

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

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

(in-package "ROS")

(defclass gazebo\_msgs\_new::GetModelPropertiesRequest

:super ros::object

:slots (\_model\_name ))

(defmethod gazebo\_msgs\_new::GetModelPropertiesRequest

(:init

(&key

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

)

(send-super :init)

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

self)

(:model\_name

(&optional \_\_model\_name)

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

(:serialization-length

()

(+

;; string \_model\_name

4 (length \_model\_name)

))

(: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)

;;

(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))

;;

self)

)

(defclass gazebo\_msgs\_new::GetModelPropertiesResponse

:super ros::object

:slots (\_parent\_model\_name \_canonical\_body\_name \_body\_names \_geom\_names \_joint\_names \_child\_model\_names \_is\_static \_success \_status\_message ))

(defmethod gazebo\_msgs\_new::GetModelPropertiesResponse

(:init

(&key

((:parent\_model\_name \_\_parent\_model\_name) "")

((:canonical\_body\_name \_\_canonical\_body\_name) "")

((:body\_names \_\_body\_names) (let (r) (dotimes (i 0) (push "" r)) r))

((:geom\_names \_\_geom\_names) (let (r) (dotimes (i 0) (push "" r)) r))

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

((:child\_model\_names \_\_child\_model\_names) (let (r) (dotimes (i 0) (push "" r)) r))

((:is\_static \_\_is\_static) nil)

((:success \_\_success) nil)

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

)

(send-super :init)

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

(setq \_canonical\_body\_name (string \_\_canonical\_body\_name))

(setq \_body\_names \_\_body\_names)

(setq \_geom\_names \_\_geom\_names)

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

(setq \_child\_model\_names \_\_child\_model\_names)

(setq \_is\_static \_\_is\_static)

(setq \_success \_\_success)

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

self)

(:parent\_model\_name

(&optional \_\_parent\_model\_name)

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

(:canonical\_body\_name

(&optional \_\_canonical\_body\_name)

(if \_\_canonical\_body\_name (setq \_canonical\_body\_name \_\_canonical\_body\_name)) \_canonical\_body\_name)

(:body\_names

(&optional \_\_body\_names)

(if \_\_body\_names (setq \_body\_names \_\_body\_names)) \_body\_names)

(:geom\_names

(&optional \_\_geom\_names)

(if \_\_geom\_names (setq \_geom\_names \_\_geom\_names)) \_geom\_names)

(:joint\_names

(&optional \_\_joint\_names)

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

(:child\_model\_names

(&optional \_\_child\_model\_names)

(if \_\_child\_model\_names (setq \_child\_model\_names \_\_child\_model\_names)) \_child\_model\_names)

(:is\_static

(&optional \_\_is\_static)

(if \_\_is\_static (setq \_is\_static \_\_is\_static)) \_is\_static)

(: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

()

(+

;; string \_parent\_model\_name

4 (length \_parent\_model\_name)

;; string \_canonical\_body\_name

4 (length \_canonical\_body\_name)

;; string[] \_body\_names

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

;; string[] \_geom\_names

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

;; string[] \_joint\_names

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

;; string[] \_child\_model\_names

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

;; bool \_is\_static

1

;; 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)))))

;; string \_parent\_model\_name

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

;; string \_canonical\_body\_name

(write-long (length \_canonical\_body\_name) s) (princ \_canonical\_body\_name s)

;; string[] \_body\_names

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

(dolist (elem \_body\_names)

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

)

;; string[] \_geom\_names

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

(dolist (elem \_geom\_names)

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

)

;; string[] \_joint\_names

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

(dolist (elem \_joint\_names)

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

)

;; string[] \_child\_model\_names

(write-long (length \_child\_model\_names) s)

(dolist (elem \_child\_model\_names)

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

)

;; bool \_is\_static

(if \_is\_static (write-byte -1 s) (write-byte 0 s))

;; 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))

;; string \_parent\_model\_name

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

;; string \_canonical\_body\_name

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

;; string[] \_body\_names

(let (n)

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

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

(dotimes (i n)

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

))

;; string[] \_geom\_names

(let (n)

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

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

(dotimes (i n)

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

))

;; string[] \_child\_model\_names

(let (n)

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

(setq \_child\_model\_names (make-list n))

(dotimes (i n)

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

))

;; bool \_is\_static

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

;; 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::GetModelProperties

:super ros::object

:slots ())

(setf (get gazebo\_msgs\_new::GetModelProperties :md5sum-) "5717f7bd34ed990fa80e28b3015027a3")

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

(setf (get gazebo\_msgs\_new::GetModelProperties :request) gazebo\_msgs\_new::GetModelPropertiesRequest)

(setf (get gazebo\_msgs\_new::GetModelProperties :response) gazebo\_msgs\_new::GetModelPropertiesResponse)

(defmethod gazebo\_msgs\_new::GetModelPropertiesRequest

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

(setf (get gazebo\_msgs\_new::GetModelPropertiesRequest :md5sum-) "5717f7bd34ed990fa80e28b3015027a3")

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

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

"string model\_name # name of Gazebo Model

---

string parent\_model\_name # parent model

string canonical\_body\_name # name of canonical body, body names are prefixed by model name, e.g. pr2::base\_link

string[] body\_names # list of bodies, body names are prefixed by model name, e.g. pr2::base\_link

string[] geom\_names # list of geoms

string[] joint\_names # list of joints attached to the model

string[] child\_model\_names # list of child models

bool is\_static # returns true if model is static

bool success # return true if get successful

string status\_message # comments if available

")

(setf (get gazebo\_msgs\_new::GetModelPropertiesResponse :md5sum-) "5717f7bd34ed990fa80e28b3015027a3")

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

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

"string model\_name # name of Gazebo Model

---

string parent\_model\_name # parent model

string canonical\_body\_name # name of canonical body, body names are prefixed by model name, e.g. pr2::base\_link

string[] body\_names # list of bodies, body names are prefixed by model name, e.g. pr2::base\_link

string[] geom\_names # list of geoms

string[] joint\_names # list of joints attached to the model

string[] child\_model\_names # list of child models

bool is\_static # returns true if model is static

bool success # return true if get successful

string status\_message # comments if available

")

(provide :gazebo\_msgs\_new/GetModelProperties "5717f7bd34ed990fa80e28b3015027a3")